



STIC Search Report

EIC 2100

STIC Database Tracking Number: 204224

TO: Tammy Nguyen
Location: RND 4D76
Art Unit: 2144
Wednesday, October 11, 2006

Case Serial Number: 09/560703

From: Lance Sealey
Location: EIC 2100
RND-4B11
Phone: 571-272-8666

Lance.Sealey@uspto.gov

Search Notes

Dear Tammy,

Here are the results of your search request.
Please let me know if you have any questions.

Lance

STIC Search Results Feedback Form

EIC 2100

Questions about the scope or the results of the search? Contact *the EIC searcher or contact:*

Alyson Dill, EIC 2100 Team Leader
272-3527, RND 4B28

Voluntary Results Feedback Form

➤ I am an examiner in Workgroup: Example: 2133

➤ Relevant prior art **found**, search results used as follows:

- ☐ 102 rejection
- ☐ 103 rejection
- ☐ Cited as being of interest.
- ☐ Helped examiner better understand the invention.
- ☐ Helped examiner better understand the state of the art in their technology.

Types of relevant prior art found:

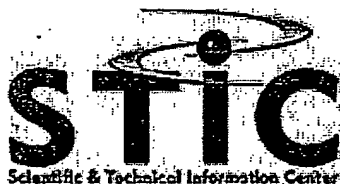
- ☐ Foreign Patent(s)
- ☐ Non-Patent Literature
(Journal articles, conference proceedings, new product announcements etc.)

➤ Relevant prior art **not found**:

- ☐ Results verified the lack of relevant prior art (helped determine patentability).
- ☐ Results were not useful in determining patentability or understanding the invention.

Comments:

Drop off or send completed forms to STIC/EIC2100 RND, 4B28



204224

STIC EIC 2100 Search Request Form

Today's Date:

10/6/06

What date would you like to use to limit the search?

Priority Date:

Other:

Name TAMU NGUYEN

AU 2144 Examiner # 79566

Room # 368 4076 Phone 3929

Serial # 09 / 560-703

Format for Search Results (Circle One):

PAPER

DISK

EMAIL

Where have you searched so far?

USP DWPI EPO JPO ACM IBM TDB

IEEE

INSPEC

SPI

Other _____

Is this a "Fast & Focused" Search Request? (Circle One) YES NO

A "Fast & Focused" Search is completed in 2-3 hours (maximum). The search must be on a very specific topic and meet certain criteria. The criteria are posted in EIC2100 and on the EIC2100 NPL Web Page at <http://ptoweb/patents/stic/stic-tc2100.htm>.

What is the topic, novelty, motivation, utility, or other specific details defining the desired focus of this search? Please include the concepts, synonyms, keywords, acronyms, definitions, strategies, and anything else that helps to describe the topic. Please attach a copy of the abstract, background, brief summary, pertinent claims and any citations of relevant art you have found.

Is this request for a BOARD of APPEALS case? (Circle One) YES NO

SEARCH PREP TIME: 32 TERMINAL TIME: 187

STIC Searcher LANCE SEALEY

Phone 2-8666

Date picked up 10/11/06

Date Completed 10/11/06



1 **ABSTRACT**

2 A Web address converter helps dynamic Web sites get the attention of
3 spiders of Internet search engines. With the Web address converter, requests from
4 Web browsers using static addresses access corresponding dynamic Web pages
5 and requests from search engines generate an instance of a Web page having links
6 with static addresses pointing to corresponding dynamic Web pages. The Web
7 address converter performs both Dynamic-to-Static (D-to-S) address conversion
8 and Static-to-Dynamic (S-to-D) address conversion. D-to-S address conversion is
9 done when generating a spider-friendly main page for a spider of a search engine
10 to crawl. S-to-D address conversion is used when a browser uses a static address
11 to access a corresponding dynamic Web page. The static address that the browser
12 uses was originally created when the spider-friendly main page was generated.
13
14
15
16
17
18
19
20
21
22
23
24
25

00/240-20/09560
09560703-1042700

CLAIM AMENDMENTS

Claim Amendment Summary

Claims pending

- At time of the Action: Claims 1-13, 23-24, 35, 39-56.
- After this Response: Claims 1, 4-13, 35, 41-45, 47, 52-54.

Canceled or Withdrawn claims: Claims 2-3, 23-24, 39-40, 46, 48-51, and 55-56.

Amended claims: Claims 1, 8, 11, 35, 41-45, 47 and 52-54.

New claims: none.

Claims:

1. **(CURRENTLY AMENDED)** A ~~spider friendly Web page~~
~~generation~~ method comprising:

receiving a request from a search engine to crawl a Website;

generating an instance of a main Web page having at least one link with a dynamic address pointing to a dynamic Web page;[[and]]

converting the dynamic address into a static address that also points to the dynamic Web page;[[.]]

sending the instance of the main Web page to the search engine;

facilitating crawling, by the search engine, the static address located on the main Web page, so that the search engine may send the static address to a user of the search engine; and

1 re-directing the user to the dynamic Web page after selection of the static
2 address.

3
4 2. (CANCELLED)

5
6 3. (CANCELLED)

7
8 4. (ORIGINAL) A method as recited in claim 1 further comprising
9 receiving a request for access to the main Web page, the request comprising a
10 static address pointing to the main Web page.

11
12 5. (ORIGINAL) A method as recited in claim 1, wherein the
13 generating comprises forming the instance of the main Web page so that the main
14 Web page contains meta-tags for facilitating indexing by a Web search engine.

15
16 6. (ORIGINAL) A method as recited in claim 1, wherein the
17 converting comprises:

18 parsing the dynamic address to identify and separate fields within the
19 dynamic address, wherein at least one field has a value; and

20 generating a static address incorporating the value of at least one field,
21 wherein the static address points to the dynamic Web page.

22
23 7. (ORIGINAL) A computer-readable storage medium having
24 computer-executable instructions that, when executed by a computer, performs the
25 method as recited in claim 1.

1 8. (CURRENTLY AMENDED) A ~~static to dynamic (S to D) Web~~
2 ~~address conversion~~ method comprising:

3 receiving a request from a search engine;

4 generating an instance of a main Web page having at least one link with a
5 dynamic address pointing to a dynamic Web page;

6 converting the dynamic address into a static address that also points to the
7 dynamic Web page;

8 sending the instance of the main Web page to the search engine;

9 facilitating crawling, by the search engine, the static address located on the
10 main Web page, so that the search engine may send the static address to a user of
11 the search engine;

12 receiving a request, from the user, for [[a]] the dynamic Web page, the user
13 request including [[a]] the static address pointing to the dynamic Web page; and

14 converting the static address to [[a]] the dynamic address also pointing to
15 the dynamic Web page.

16
17 9. (ORIGINAL) A method as recited in claim 8 further comprising
18 providing the dynamic address to a server.

19
20 10. (ORIGINAL) A method as recited in claim 8 further comprising
21 invoking the dynamic Web page referenced by the dynamic address.

22
23 11. (CURRENTLY AMENDED) A method as recited in claim 8
24 further comprising sending the dynamic Web page referenced by the dynamic
25 address to ~~a requester~~ the user.

12. (PREVIOUSLY PRESENTED) A method as recited in claim 8,
wherein the converting comprises:

parsing the static address to identify at least one value associated with a
field within the static address; and

generating a dynamic address incorporating at least one value associated
with the field, wherein the dynamic address points to the dynamic Web page.

13. (ORIGINAL) A computer-readable storage medium having
computer-executable instructions that, when executed by a computer, performs the
method as recited in claim 8.

14-34. (CANCELLED)

35. (CURRENTLY AMENDED) A method of providing a dynamic
Web page comprising:

receiving a request from a search engine to crawl a Website;

generating an instance of a main Web page having at least one link with a
dynamic address pointing to a dynamic Web page;

converting the dynamic address into a static address that also points to the
dynamic Web page;

facilitating crawling, by the search engine, the static address located
on the main Web page, so that the search engine may send the static
address to a computer on a network;

1 receiving a request for [[a]]the dynamic Web page from [[a]]the computer
2 on [[a]]the network, the request including [[a]]the static Web address pointing to
3 the dynamic Web page;

4 generating an instance of the dynamic Web page such that contents of the
5 instance appears as a static Web page; and

6 sending the dynamic Web page to the computer.

7
8 36-40. (CANCELLED)
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9266
F: 509.323-8979
www.lee@hayes.com
lee@hayes

1 41. (CURRENTLY AMENDED) A Web site system comprising:
2 a Web server hosting a dynamic Web site and facilitated to provide access
3 to a search engine;
4 a database storing data used by the Web server to generate dynamic Web
5 pages of the dynamic Web site, the database being operatively coupled to the Web
6 server; [[and]]
7 a dynamic to static (D-to-S) Web address converter, the converter being
8 operatively coupled to the Web server, the D-to-S Web address converter being
9 configured to convert a dynamic address pointing to a dynamic Web page into a
10 static address also pointing to the dynamic Web page in order to allow the search
11 engine to crawl the Web site; and
12 a static to dynamic (S-to-D) Web address converter, the converter being
13 operatively coupled to the Web server[[;]], the S-to-D Web address converter
14 being configured to convert a static address to a dynamic address pointing to a
15 dynamic Web page in response to a user selecting the static address as provided to
16 the user by the search engine.

1 42. (CURRENTLY AMENDED) A Web site system comprising:
2 a Web server hosting a dynamic Web site;
3 a database storing data used by the Web server to generate dynamic Web
4 pages of the dynamic Web site, the database being operatively coupled to the Web
5 server; and
6 a dynamic to static (D-to-S) Web address converter, the converter being
7 operatively coupled to the Web server;
8 the D-to-S Web address converter being configured to convert a dynamic
9 address pointing to a dynamic Web page into a static address also pointing to the
10 dynamic Web page; and
11 wherein the system is configured to create an instance of a main Web page
12 containing the converted static address, in response to a request from a search
13 engine to crawl the Web site, and is configured to facilitate crawling, by the search
14 engine, the static address located on the main Web page, so that the search engine
15 may send the static address to a user of the search engine.

1 43. (CURRENTLY AMENDED) A server comprising:
2 a processor;
3 a request receiver executable on the processor to receive a request, from a
4 search engine, including a static address of a main Web page;
5 a spider-friendly Web page generator executable on the processor to:
6 receive the static address of the main Web page from the request
7 receiver;
8 in response to receiving the static address, generate an instance of the
9 main Web page having at least one link with an address pointing to a
10 dynamic Web page; and
11 send the instance of the main Web page to the search engine.
12
13
14
15
16
17
18
19
20
21
22
23
24
25

1 44. (CURRENTLY AMENDED) A server comprising:

2 a processor;

3 a dynamic to static (D-to-S) Web address converter executable on the

4 processor to:

5 convert a dynamic address pointing to a dynamic Web page into a
6 static address also pointing to the dynamic Web page; and

7 generate an instance of a main Web page containing the static
8 address in order to facilitate crawling by a search engine;

9 a static to dynamic (S-to-D) Web address converter executable on the
10 processor to:

11 convert [[a]] the static address pointing to [[a]] the dynamic Web
12 page into [[a]] the dynamic address that also points to the dynamic Web
13 page;

14 wherein the S-to-D Web address converter converts the static
15 address in response to selection of the static address by a user of the search
16 engine.

1 45. (CURRENTLY AMENDED) A server comprising:

2 a processor;

3 a dynamic to static (D-to-S) Web address converter executable on the

4 processor to:

5 convert a dynamic address pointing to a dynamic Web page into a
6 static address also pointing to the dynamic Web page; and

7 generate an instance of a main Web page containing the static
8 address in order to facilitate crawling by a search engine;

9 a static to dynamic (S-to-D) Web address converter executable on the
10 processor to:

11 parse [[a]] the static address to identify at least one value associated
12 with a field within the static address; and

13 generating [[a]] the dynamic address incorporating at least one value
14 associated with the field, wherein the dynamic address points to the
15 dynamic Web page;

16 wherein the S-to-D Web address converter parses and generates in
17 response to selection of the static address by a user of the search engine.

18
19 46. (CANCELLED)

1 47. (CURRENTLY AMENDED) A system for hosting dynamic Web
2 sites comprising:

3 a Web server for dynamically generating an instance of a dynamic Web
4 page; and

5 a spider-friendly Web page generator configured to:

6 generate, in response to a request from a search engine spider to crawl
7 the Web site, an instance of a main Web page having at least one link with
8 a dynamic address pointing to a dynamic Web page; [[and]]

9 convert the dynamic address into a static address that also points to the
10 dynamic Web page;[[.]]

11 send the instance of the main Web page to the search engine spider in
12 order to facilitate crawling of the Web site by the search engine spider; and
13 a static to dynamic (S-to-D) Web address converter to re-direct a user of the
14 search engine to the dynamic Web page after selection of the static address.

15
16 48-51. (CANCELLED)

17
18 52. (CURRENTLY AMENDED) A computer-readable storage
19 medium having computer-executable instructions that, when executed by a
20 computer, performs a spider-friendly Web page generation method comprising:

21 receiving a request from a search engine to crawl a Website;
22
23
24
25

generating an instance of a spider-friendly Web page having at least one link with a dynamic address pointing to a dynamic Web page; [[and]]

converting the dynamic address into a static address that also points to the dynamic Web page;[[.]]

sending the instance of the spider-friendly Web page to the search engine;

facilitating crawling, by the search engine, the static address located on the spider-friendly Web page, so that the search engine may send the static address to a user of the search engine; and

re-directing the user to the dynamic Web page after selection of the static address.

53. (CURRENTLY AMENDED) A computer-readable storage medium having computer-executable instructions that, when executed by a computer, performs a ~~static to dynamic (S to D) Web address conversion~~ method comprising:

receiving a request from a search engine to crawl a Website;

generating an instance of a spider-friendly Web page having at least one link with a dynamic address pointing to a dynamic Web page;

converting the dynamic address into a static address that also points to the dynamic Web page;

sending the instance of the spider-friendly Web page to the search engine;

facilitating crawling, by the search engine, the static address located on the spider-friendly Web page, so that the search engine may send the static address to a user of the search engine;

1 receiving a request for ~~[[a]]the~~ dynamic Web page, wherein the request
2 includes ~~[[a]]the~~ static address pointing to the dynamic Web page; and
3 converting the static address to ~~[[a]]the~~ dynamic address that also points to
4 the dynamic Web page.

5
6 54. (CURRENTLY AMENDED) ~~A computer readable storage~~
7 ~~medium having computer executable instructions that, when executed by a~~
8 ~~computer, performs a static to dynamic (S to D) Web address conversion method~~
9 ~~comprising: The computer-readable storage medium of claim 53 having computer-~~
10 ~~executable instructions further comprising:~~

11 ~~receiving a static address pointing to a dynamic Web page;~~
12 ~~parsing the static address to identify at least one value associated with a~~
13 ~~field within the static address; and~~
14 ~~generating a dynamic address incorporating at least one value associated~~
15 ~~with the field, wherein the dynamic address points to the dynamic Web page.~~

16
17 55-56. (CANCELLED)
18
19
20
21
22
23
24
25

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.lee@hayes.com
lee@hayes

Set	Items	Description
S1	1298443	(RETRIEV??? OR IDENTIF???????? OR LOCAT??? OR SEARCH??? OR - QUERY??? OR CRAWL??? OR TRAVERS??? OR WALK???)(10N)(INTERNET - OR INTRANET OR (INTER OR INTRA OR EXTRA)())NET OR EXTRANET? ? - OR WWW OR WEB OR PORTAL? ? OR ONLINE? OR ON()LINE?)
S2	1060855	(REDIRECT??? OR TRANSMIT??? OR TRANSMIS???? OR RETRANSMIT?- ?? OR RETRANSMIS???? OR TRANSFER??? OR FORWARD??? OR ROUT??? - OR MIGRAT??? OR SEND??? OR DISPATCH??? OR SENT OR RESEND??? - OR RESENT)(5N)(USER??? OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR BUYER? ?
S3	15	S2(5N)(STATIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR RESOURCE()LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOME- PAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ? OR SITE? ?)-))
S4	0	S3(5N)(DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR UR- L? ? OR RESOURCE()LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HO- MEPAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ? OR SITE? - ?)) OR ASP? ? OR ACTIVE()SERVER()PAGE? ? OR JAF? ? OR JAVABEA- NS()ACTIVATION
S5	4803	STATIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR - RESOURCE()LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ? OR SITE? ?))
S6	533347	DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR RESOURCE()LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ? OR SITE? ?)) OR AS- P? ? OR ACTIVE()SERVER()PAGE? ? OR JAF? ? OR JAVABEANS()ACTIV- ATION()FRAME
S7	230	S2 AND S5 AND S6
S8	89	S7 AND (PY<2000 OR PD<19990427)
S9	61	RD (unique items)

? show files

File 275:Gale Group Computer DB(TM) 1983-2006/Oct 10
(c) 2006 The Gale Group

File 47:Gale Group Magazine DB(TM) 1959-2006/Oct 10
(c) 2006 The Gale group

File 16:Gale Group PROMT(R) 1990-2006/Oct 10
(c) 2006 The Gale Group

File 624:McGraw-Hill Publications 1985-2006/Oct 11
(c) 2006 McGraw-Hill Co. Inc

File 484:Periodical Abs Plustext 1986-2006/Oct W1
(c) 2006 ProQuest

File 613:PR Newswire 1999-2006/Oct 11
(c) 2006 PR Newswire Association Inc

File 813:PR Newswire 1987-1999/Apr 30
(c) 1999 PR Newswire Association Inc

File 239:Mathsci 1940-2006/Nov
(c) 2006 American Mathematical Society

File 370:Science 1996-1999/Jul W3
(c) 1999 AAAS

File 696:DIALOG Telecom. Newsletters 1995-2006/Oct 10
(c) 2006 Dialog

File 621:Gale Group New Prod.Annou.(R) 1985-2006/Oct 10
(c) 2006 The Gale Group

File 674:Computer News Fulltext 1989-2006/Sep W1
(c) 2006 IDG Communications

File 88:Gale Group Business A.R.T.S. 1976-2006/Oct 10
(c) 2006 The Gale Group

File 369:New Scientist 1994-2006/Aug W3
(c) 2006 Reed Business Information Ltd.

File 160:Gale Group PROMT(R) 1972-1989
(c) 1999 The Gale Group

File 635:Business Dateline(R) 1985-2006/Oct 11
(c) 2006 ProQuest Info&Learning
File 15:ABI/Inform(R) 1971-2006/Oct 11
(c) 2006 ProQuest Info&Learning
File 9:Business & Industry(R) Jul/1994-2006/Oct 10
(c) 2006 The Gale Group
File 13:BAMP 2006/Oct W1
(c) 2006 The Gale Group
File 810:Business Wire 1986-1999/Feb 28
(c) 1999 Business Wire
File 610:Business Wire 1999-2006/Oct 11
(c) 2006 Business Wire.
File 647:CMP Computer Fulltext 1988-2006/Nov W4
(c) 2006 CMP Media, LLC
File 98:General Sci Abs 1984-2006/Oct
(c) 2006 The HW Wilson Co.
File 148:Gale Group Trade & Industry DB 1976-2006/Oct 11
(c)2006 The Gale Group
File 634:San Jose Mercury Jun 1985-2006/Oct 10
(c) 2006 San Jose Mercury News
File 636:Gale Group Newsletter DB(TM) 1987-2006/Oct 10
(c) 2006 The Gale Group

?

9/9/1 (Item 1 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2006 The Gale Group. All rts. reserv.

02359607 SUPPLIER NUMBER: 58326330 (THIS IS THE FULL TEXT)
Making your site interactive. (Microsoft's Active Server Pages
) (Technology Tutorial)
Reynolds, Matthew
Internet Magazine, 115
Nov, 1999
ISSN: 1355-6428 LANGUAGE: English RECORD TYPE: Fulltext
WORD COUNT: 2988 LINE COUNT: 00241

TEXT:

The difference between flat, static Web pages and interactive, dynamic ones is Active Server Pages (ASP). Matthew Reynolds tells you how to add ASP to your pages.

There seems to be no limit to what you can do online now. You can buy books and videos, check on cinema listings and stocks and shares, sell the junk in your garage through auction sites, and send email to your friends and family. So how did we get from only being able to read flat, scientific papers to all these other cool things? The answer is simple -- Web sites can now react and interact with you to perform tasks.

Welcome to Yahoo!

The first successful dynamic Web site was Yahoo!. It works by using a system called CGI, or Common Gateway Interface. Yahoo! wrote software that could trawl a massive database of Web site links and return the search results back to your browser.

The search engine is driven by CGI technology that lets dynamic Web sites present database queries to their visitors. The world's top 50 sites are all dynamic.

At the data centre -- the room that houses the servers that run the Site, as well as the administration team -- a Web server is configured to run CGI applications. The Web server interprets URLs that are requested by browsers.

If a URL has a file name that CGI understands, the Web server sends the request to another process dedicated to CGI processing. This process uses the code contained within the page to generate HTML code that's sent back to the browser.

CGI is configured to interpret the file types .pl, .exe and .dll. You can tell if a site is running CGI by looking for these extensions in its URL in the address bar.

The software that powers the Web server's CGI abilities runs on an application server. The functionality of a Web site is often divided into three parts -- a Web server (sometimes called an HTTP server), an application server and a database server.

This division helps the site deal with more simultaneous clients. Usually the Web server sends out static information, such as images and HTML pages, while the application and database servers generate dynamic content.

CGI is a Unix technology -- it never really took off on Windows NT. But Microsoft invented something arguably better than CGI -- Active Server Pages.

Active Server Pages

Active Server Pages (ASP) are Microsoft's answer to the growing need to create dynamic Web sites.

Like most things in the Unix world, CGI can be difficult to set up and use. The de facto language for writing CGI applications, Perl, is harder to use than Visual Basic (the basic programming language Microsoft created for developing Windows applications). Visual Basic accounts for 70

percent of all the world's desktop software, so it was logical for Microsoft to base ASP on the Visual Basic language.

You can pick and choose the language you want to use with Visual Basic, but most people opt for a subset of the programming language called VBScript. Microsoft deliberately made ASP more accessible than CGI in order to make developers base their Web applications on the Windows NT platform. In short, if you know how to write code in Visual Basic, you'll be building dynamic Web applications within minutes.

When to use ASP

Even if you're not planning to build an e-commerce site from scratch, or outdo eBay, you can still use ASP.

For instance, you'll find it useful when your Web site has to connect to a database of some sort. ASP is capable of connecting to most databases but it's optimised to converse with SQL Server 7, the new Microsoft Data Engine (MSDE) and MS Access.

There are two places where an organisation can use dynamic Web sites. The first is to build an intranet, which is a Web site that can only be accessed by people inside your company. You might use your company intranet to let people look at customer records, place orders or book holiday time. ASP is perfect for these applications.

The second place you can use ASP is on your company Web site. Even if it's just a flat site with pictures of your products and some text, you can still use ASP to provide a customer feedback form, a survey or other tools that'll help your customers interact with your organisation. You can configure these forms to write the results directly into a database, email the results to the relevant parties, or create text files on the server.

Whatever you want to do with ASP, you need to make sure you have somewhere to host the site. Most people use a commercial hosting service, rather than hosting in-house. You'll need to make sure your host will support ASP, but most do. ASP hosting costs from (pounds)25 a month for low-end hosting (the level you'd need for a customer feedback form) to several hundred pounds per month for high-end hosting - the level you'll need for a complete e-commerce site.

If you're doing your hosting in-house on Windows NT, you'll have no problems hosting ASP. After you've done your development work, make sure your production server is configured to run ASP and to copy your new code.

Using ASP

ASP is available on all Windows platforms, from Windows 95 to an NT 4.0 Server. But you don't have to be running Windows to see a site with ASP. ASP code generates HTML, so it can be read by anyone.

If you're building sites with FrontPage 98, 99 or 2000, you're ready to start building ASP pages. If you're not using FrontPage, read the Where to find ASP panel, which tells you how to install ASP.

The Personal Web Server will have been installed with FrontPage, along with the appropriate server extensions you need to get working.

You can build your ASP pages in any text editor you choose. Most serious ASP developers use a Microsoft Visual Studio product called Visual InterDev, but there's no reason why you can't use Notepad, a shareware text editor or FrontPage.

Visual InterDev is the preferred tool for ASP developers because it offers some cool tools and can communicate with Web servers. This makes it easier for you to deploy your ASP code to the server.

Eitherway, building ASP pages is incredibly easy. An ASP page is simply an HTML page with a .asp extension and code embedded in it. You can find ASP-driven sites by looking for .asp in the URL.

As in CGI, a Web server will look for requests ending in .asp. When it receives a request, it passes the processing of that request to DLL (Dynamic Link Library), called asp.dll, and it's the responsibility of this DLL to turn the ASP request into HTML code that the browser can

understand.

Here's a simple HTML page before we make it active with some ASP code:

```
(less than)HTML(greater than)
(less than)HEAD(greater than)
(less than)TITLE(greater than)My ASP Page(less than)/TITLE(greater
than)
(less than)/HEAD(greater than)
(less than)BODY(greater than)
Hello World!
(less than)/BODY(greater than)
(less than)/HTML(greater than)
```

To make our static, boring HTML page into a dynamic, exciting ASP page, you simply add some VBScript code to the page. You tell ASP where your code starts and ends using the special (less than)% and %(greater than) tags. Here's how you can change your ASP code, so instead of Hello World!, you can tell your visitors the result of the tricky scientific calculation $2 + 2$:

```
(less than)HTML(greater than)
(less than)HEAD(greater than)
(less than)TITLE(greater than)My ASP Page(less than)/TITLE(greater
than)
(less than)/HEAD(greater than)
(less than)BODY(greater than)
Here's my tricky calculation: (less than)%=2 + 2%(greater than)
(less than)/BODY(greater than)
(less than)/HTML(greater than)
```

As soon as the (less than)% and %(greater than) tags are found, ASP knows it's supposed to execute whatever code is contained within them. The equals sign tells ASP to send the results of the code back to the visitor's browser. In this example, ASP works out what $2 + 2$ is, and sends the answer back as 4.

Building applications

One of the best things about ASP is its open architecture. Microsoft is keen to make sure its technologies are easy to access and use. Its favourite way of gluing things together is using ActiveX -- a way of plugging components into containers.

The majority of traditional desktop applications can access a database. As you'd expect, the majority of dynamic Web applications can do it too. The technology most commonly used for integrating databases into ASP sites is ActiveX Data Objects (ADO). ADO is an ActiveX software component and ASP is an ActiveX container.

At any point in executing an ASP page, five ActiveX objects are always available to the code that runs in that page. These objects -- request, response, application, session and server -- provide ways for the ASP page to communicate with the server environment and the visitor's browser. They're known as ASP built-in objects, or ASP intrinsic objects.

The main function of the server component is to create instances of other ActiveX components. So you could use the Server component to create an ADO object capable of connecting to a database, and use that ADO object to connect.

Here's how you can create, and connect to, an SQL Sever database running on your network:

```
(less than)HTML(greater than)
(less than)HEAD(greater than)
(less than)TITLE(greater than)My ASP Page(less than)/TITLE(greater
than)
(less than)/HEAD(greater than)
(less than)%
```

```

        (less than)BODY(greater than)
        Create an ADO Connection object...
        Set DB = Server.CreateObject
        ("ADODB.Connection") Connect that object to the database... DB.Open
"driver=SQL Server;database=DB;uid=SA;pwd=;server=DBSERVER"
        %(greater than)

```

(less than)/BODY(greater than)
 (less than)/HTML(greater than)
 ADO (ActiveX Data Objects) is an excellent collection of objects that lets you do almost anything you want with a database. Once you've established a connection, you can carry out tasks, such as producing a list of your customers or viewing the orders you've received.

Here's how easy it is to produce a list of your customers back and display them to your visitors:

```

        (less than)HTML(greater than)
        (less than)HEAD(greater than)
        (less than)TITLE(greater than)My ASP Page(less than)/TITLE(greater
than)

```

```

        (less than)/HEAD(greater than)
        (less than)BODY(greater than)
        (less than)%
        Create an ADO Connection object...
        Set DB = Server.CreateObject ("ADODB.Connection")
        Connect that object to the database...
        DB.Open "driver=sql server;database=db;uid=sa;pwd=;server=dbserver"
        Open a list of the customers...
        Set Query = DB.Execute("select * from customers")
        Loop the customers ...
        Do While Not Query.EOF
            Send the customer details back to the visitor...
            Response.Write Query("name") & " " & Query("email") & "(less
than)br(greater than)"
            Response.Write Query ("company") & "(less than)br(greater than)"
            Response.Write "(less than)br(greater than)"
            Next...
            Query.MoveNext
            %(greater than)
        Loop
        (less than)/BODY(greater than)
        (less than)/HTML(greater than)

```

The Response.Write calls use the built-in response object. The main job of this object is to send HTML-formatted text back to the browser through the Response.Write call. In the example, it's displaying the customer information we've taken from the database using ADO.

Third-party components

Thanks to the open nature of ASP, there's a wealth of third-party components you can install on your server and use in your application.

One of the most common requirements of an active site is the ability to send emails, and you can get a number of components that can do that. My particular favourite is Dimac's Mail component (www.dimac.net). Another popular component is Server Object's ASPMail (www.serverobjects.com).

You can download components from the ASP Resource Index (at www.aspin.com) that can ping devices on your network, integrate with e-commerce Payment Service Providers and present data as graphs to your site's visitors. And there's a host of cool things you can make your new active site do.

Summary

The list of ASP-driven sites is growing every day, as Microsoft continues to play catch-up in the Web platform race. ASP development is substantially easier and more accessible than CGI development, which (in a number of expert opinions) means Microsoft could be more successful than

Unix and CGI, just as it appears to have beaten Netscape on the desktop.

In the upcoming release of Windows 2000, Microsoft will be including ASP 3.0. This new version includes a number of small changes that developers have wanted for a longtime. The bulk of the work addresses concerns from professional Web developers that NT is not up to the task of hosting sites that attract a high amount of traffic. This is proof that Microsoft wants NT, IIS and ASP to be the de facto dynamic Web hosting standard as businesses continue to grow and adapt to the challenges of doing business online.

Matthew Reynolds is senior editor

Where to find ASP

Win95/98 and NT Workstation users

You'll need the Personal Web Server, also known as PWS, which you can download for free from the Microsoft Web site (www.microsoft.com), or you can find it in the Microsoft Windows NT Option Pack. Windows 95 and 98 users can successfully install the Personal Web Server from the NT Option pack. Either way, make sure you install the FrontPage 98 Server Extensions. This contains the ASP components, but also the components that let your developers put their work directly onto your servers, using Visual InterDev.

Windows NT Server users

You already have ASP installed on your computer along with Internet Information Server (IIS). Make sure you also have FrontPage 98 Server Extensions installed.

Create a customer feedback form in ASP

If you have FrontPage 98 or FrontPage 2000 installed on your computer, you can start writing ASP pages straight away. You can use any text editor to build ASP pages, though, such as Notepad, a shareware text editor, or Visual InterDev. FrontPage 2000 comes with built-in wizards to build feedback forms for you, but you'll develop a sound understanding of ASP if you build one yourself

Step 1

After opening the FrontPage Web you want to edit, create a new page. Design a form similar to the one you see here. The floating boxes tell you the name of each field. To set the name, right-click on each field, choose Field Properties and then enter the name. Then save the page you've created as feedbackform.htm.

Step 2

Now we need to configure the form to post the contents of the form to another ASP page. This is a fairly typically approach to using ASP - build a single HTML page that presents the form to the user, and then build an ASP that processes the data the visitor enters into the form. To make the form run another ASP page, right-click on the form, select Form Properties, check on Send to other and make sure Custom ISAPI, NSAPI, CGI or ASP Script is selected. Click the options button at the bottom of the dialog and set the action field to processform.asp.

Step 3

Our next step is to create the "processform.asp" page. To do this, just create a new page, click the "Save" button and enter its name as "processform.asp". You'll then need to enter the ASP code that will take the contents of the form, open a text file on the server and write the visitor's name, e-mail address and comments into that file. Here's the code that will do that for you:

```
(less than)html(greater than)
(less than)head(greater than)
(less than)meta http-equiv="Content-Type" content="text/html;
charset=windows-1252"(greater than)
(less than)meta name="GENERATOR" content="Microsoft FrontPage
4.0"(greater than)
(less than)meta name="ProgId"
```



```

content="FrontPage.Editor.Document"(greater than)
(less than)title(greater than)New Page 1(less than)/title(greater
than)
(less than)/head(greater than)
(less than)body(greater than)
(less than)%
Create a customer feedback form in ASP
Firstly, open a text file on the server to write the values into. The
Server.MapPath command turns a virtual Web path into a physical folder on
disk. In this case, Server.MapPath ("/") will return the folder as
C:\inetpub\wwwroot...
filename = Server.MapPath("/") & "\feedback.txt"
We can write the name of the file back to the browser (not that you'd
ever want to do this in the `real world'!), using the Response.Write
method...
Set FSO = Server.CreateObject("Scripting.FileSystemObject")
Set file = FSO.OpenTextFile(filename, 8, True)
Response.Write "We're saving in: "& filename & "(less than)br(greater
than)(less than)br(greater than)"
Secondly, we need to use the built-in objects that will Let us create
text files. To do this, we create an instance of the ActiveX component
Scripting.FileSystemObject, then open a text file...
Thirdly, we can use the ASP built-in object Request to get the data
out of the form by passing the name of the field we want to the Request
object's Form property. Then, we write each one as a separate line in the
file...
file.WriteLine "Comments received at: " & Now
file.WriteLine Request.Form("email")
file.WriteLine Request.Form("name")
file.WriteLine Request.Form("comments")
file.WriteLine 'add a blank line at the end
Lastly, we need to clean up everything we've opened...
file.close
Set File = Nothing
Set FSO = Nothing
%(greater than)
Form Processed. Congratulations! :-)
(less than)/body(greater than)
(less than)/html(greater than)
Step 4
With all the code in place, you can fill out your form:
Step 5
When the Submit button is pressed, the browser will pass the contents
of the form over to the page defined as the ACTION page for the form, in
this case processform. asp . The code we added to "processform. asp " will
process the contents of the form, adding the comments to a text file on the
server which we can read in Notepad.

```

COPYRIGHT 1999 EMAP Media Ltd.

COMPANY NAMES: Microsoft Corp.--Products
 GEOGRAPHIC CODES/NAMES: 1USA United States
 DESCRIPTORS: Programming tutorial; Web site/Web page development;
 Internet/Web server software
 EVENT CODES/NAMES: 330 Product information
 PRODUCT/INDUSTRY NAMES: 7372682 (Internet Server Software)
 NAICS CODES: 51121 Software Publishers
 TRADE NAMES: Microsoft Active Server Pages (Internet/Web server
 software)--Usage

9/9/3 (Item 3 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2006 The Gale Group. All rts. reserv.

02309515 SUPPLIER NUMBER: 55022198 (THIS IS THE FULL TEXT)
Control Web Site Content. (WebTrends' WebTrends Log Analyzer, WebTrends
Enterprise Suite;Mercury Interactive's Astra Site Manager, InContext
Systems' WebAnalyzer 2.01, Tetranet Software's Linkbot Pro 4.0,
Electronic Software Publishing's LinkScan, other Web site analysis
software) (Software Review) (Evaluation)

Steinke, Steve

Network, NA

June 1, 1999

DOCUMENT TYPE: Evaluation ISSN: 1093-8001 LANGUAGE: English

RECORD TYPE: Fulltext; Abstract

WORD COUNT: 3536 LINE COUNT: 00284

ABSTRACT: WebTrends' WebTrends Enterprise Suite includes multiple modules for analyzing log files, proxy server usage, streaming media performance and monitor availability. WebTrends' \$499 Site Manager can both access sites and block access to sites via proxy servers, ensuring an extra level of security. Mercury Interactive's \$495 Astra Site Manager generates structural diagrams of Web sites during scanning, revealing bad links and access problems on the fly.

TEXT:

If your Web site is part of your business, you're obliged to maintain the reliability, efficiency, and attractive- ness of its operation to at least the same degree as you would your physical facilities. The online equivalents of long checkout lines, slow service, absence of help, missing information, and out-of-stock products are every bit as frustrating to customers as their physical-world counterparts.

Some approaches to Web site management focus on the network and on network- related performance enhancement. Some of these classes of products include server load balancing, bandwidth management, and caching. When the capacity of the network pipe or the raw performance of Web servers can be identified as the principal bottleneck to timely Web content delivery, these technical solutions can clearly make a difference. This article won't pursue these approaches, though we have covered them previously ("Eliminating Web Server Bottlenecks," April 1998, page 50; "Managing Bandwidth to Allocate Scarce Resources," November 1998, page 64; "Tutorial 129: Proxy Servers," April 1999, page 27).

Other approaches focus on managing the content of a Web site. One method involves the use of all-in-one site devel- opment tools such as LivePage Enterprise, Vignette StoryServer, or Allaire's ColdFusion. While these types of products provide various forms of integrated content management, they require complete commitment to their own way of doing things, which can conflict with the design of existing sites. The pros and cons of these large-scale integrated environments are also beyond the scope of this article.

What this article will cover are the products that have the ability to detect and possibly correct content-based problems that detract from the quality of a Web site.

QUALITY CONTROL

We often gauge the quality of a Web site by how valuable the information is, by how well written the text is, or by the overall level of competence the site displays in its text, images, graphics, and sounds. Unfortunately, there are no tools that can detect good content in this sense-and some would argue that such a tool is impossible in principle.

The kinds of quality indicators that software can detect have to do with speed, correctness, and accessibility. Here, speed refers to the results from content-based choices as opposed to speed based on the

throughput of the network or the Web server software. For example, a design based on frames that doesn't display anything until the last file is loaded into the browser will invariably be perceived to be slower than a design that gives the user's browser something it can display immediately.

Download delays are consistently the biggest complaint users have about Web sites. Web content needs to take a sort of Hippocratic oath-"Do nothing to interfere with download speed." File size is a primary enemy of speed, and will continue to be important as long as a significant fraction of users has sub-ISDN rate (that is, less than 64Kbit/sec) Internet access. It seems to be widely agreed now that downloading Java executables to a browser for the purpose of animating text (or doing much of anything else that doesn't accomplish meaningful work that can't be done some other way) is a waste of limited resources. Many of the tools I'll examine can present lists of files ordered by size and type, readily identifying the biggest obstacles to speed resulting from the site's content.

The loss of correctness has two primary culprits-broken links and broken pages. Broken links can include same-site URLs and other-site URLs, but in either case, the user gets a 404 File not found message or some modernized equivalent, along with a strong impression of a sloppy, unprofessional Web site. Broken pages are generally HTML coding errors that may result in display problems as minor as an HTML tag displayed in a place it doesn't belong, or as major as a page that fails to load at all. From the user's perspective, most broken pages are just as disturbing as broken links.

Broken links can be introduced in a multitude of ways. One common cause is changing a file name. Also, changing a directory name incorrectly can result in dozens, if not hundreds, of broken links. Moving files and directories, or inserting new levels of directories, can also profoundly disrupt a site. Any file deletion is a potential broken link. Links to other sites are particularly vulnerable-the other site's administrators may not know your site links to them, but even if they did, they might not feel much of an obligation to notify you when they make improvements and modifications to their Web site.

Fortunately, broken links are the sort of problem that programmers find easy to solve. Each of the numerous link-checking software packages sails through the source code on a starting Web page, finds all the links, and jumps to them one after another. On each successive page it jumps to, the software finds all the links and jumps to them. Along the way, it keeps track of the errors it finds and writes them to a file or a database so it can report them. Simple enhancements include the ability to stop at the boundaries of the local site, to limit the number of jumps on local or remote sites, and to filter certain sorts of URLs, such as excluding CGI calls or including only streaming media URLs or only particular subdirectories.

If a program is systematically traversing all the pages in a site via their links, it would be a pity to refrain from generating a map or diagram of the site. In fact, a number of the broken link detectors have mundane or ambitious displays of complete Web site structures, often with broken links highlighted.

HTML analysis is mostly limited to syntax checking. While this operation is capable of identifying many troubled pages on a Web site, there are other potential errors, such as mistaken tags and badly chosen fonts, that can only be detected with a visual inspection. Hopefully the advance of XML and the clear separation of formatting instructions from content will also result in fewer formatting and layout errors.

ACCESSIBILITY FACTORS

Accessibility is a widely admired property of Web sites, but a difficult one to pin down-especially for a program. The Web-Criteria people (www.webcriteria.com), who provide a comprehensive Web site evaluation service (see "Outsourcing Web Site Quality," page 63), define accessibility as the time it takes for a user to get to different parts of

a Web site. The evaluation includes an examination of all of the potential paths to each page of the site, with varying weights based on how big the data content of the pages are and how much of that data is repeated (and therefore potentially cached) as the user moves from page to page. This technique may represent the best measure that can be expected of an automated process.

However, there is certainly a sense of accessibility that is based on the semantic content of the site. For example, a corporate home page that doesn't have equivalents of "About the Company," "Products and Services," and "Contact Us" as beginning navigation points is asking for confused users. There's no way a program can uncover the ideal logical division of subject matter for clear site navigation.

Some of the products in this article combine site mapping with the ability to capture traffic patterns. Reports based on this sort of information could be useful for enhancing accessibility by helping the site designer simplify access to the most popular parts of the site. Of course, broken links and broken pages hamper accessibility in addition to their other unpleasant properties.

CONTENT MANAGEMENT PRODUCTS

The market for content management is still a youthful, volatile one. Some of the products are shareware or freeware, available only by downloading them from the Internet. Others began on a shoestring and grew into high-end enterprise tool suites. Site Technologies, along with its SiteSweeper product that seemed to be a leading contender in this category a year ago, has apparently vanished from the face of the earth. One of the early site mapping and link checking products was WebMapper, from NetCarta. But Microsoft acquired NetCarta in late 1996, and if any of the WebMapper features survived, they are submerged in Front Page or IIS.

One of the best known names in Web site management is WebTrends (www.webtrends.com), whose WebTrends Log Analyzer is a widely used traffic analysis and reporting tool. WebTrends Enterprise Suite includes modules or "cartridges" that can analyze proxy server usage, monitor availability, analyze streaming media performance, and report on the performance of server clusters, as well as analyze log files for traffic reporting. A cartridge named Site Manager is the primary tool for managing site content.

Site Manager finds broken links and broken pages. It identifies the largest and slowest-to-load pages. It can identify the newest and oldest pages, and create a catalog of image files for review. Site Manager can be integrated with editing tools so that repairing links or other problems becomes a single-step process. A feature of Site Manager that many large enterprises need is the ability to access sites through a proxy server or to block access behind a proxy server.

Site Manager has five display modes, each of which emphasizes particular features of a site's structure. The file-type view arranges the site components in an Explorer-like hierarchy arranged by the type of content they contain. The link-type view is a list of all the links in the site, identifying the files each link points to. The interactive graphical Web chart draws lines between icons that represent the site components, emphasizing the graphical structure of the site. The interactive hyperbolic tree is hard to explain, and even an image (see Figure 1, page 61) fails to do it justice. Suffice it to say that it is a clever way to present very dense graphical objects in an understandable and intuitive two-dimensional display. Finally, there is a group view, which classes together broken links, internal links, external links, system errors, and other sorts of links. Site Manager costs \$499.

Another product that aims to meet the needs of the enterprise market is Mercury Interactive's (www.merc-int.com) Astra Site Manager, which generates structural diagrams of Web sites as it scans them, indicating broken links and access problems as it goes. It can also highlight differences in the site from one scan to another, a valuable way to check that any changes you make are the ones you intend. Astra can scan dynamic

pages as well as static ones by inserting queries into CGI forms in advance.

Astra Site Manager's viewing options include Visual Web Display, Action Tracker, Link Doctor, Change Viewer, and Dynamic Scanner. Visual Web Display is the architectural overview that includes all of the site's objects. Action Tracker monitors and captures usage patterns on the site. Link Doctor presents all the broken links and provides integrated access to your chosen editor so they can be repaired. Change Viewer compares site maps over time, and Dynamic Scanner, which generates pages on the fly with CGI requests, makes it possible to check the display of database and transaction-oriented operations. Mercury Interactive's product costs \$495.

While WebAnalyzer 2.01, from InContext Systems (www.incontext.com), offers a variety of the usual hierarchical list displays of links and URLs, its Wavefront View is unique in this class of products (see Figure 2). Beginning with the home page, each page that's one link away is arranged in a circle around the home page; each page two links away is arranged in a concentric circle around the one-link circle; and so on to an arbitrary number of levels. You can zoom in on the Wavefront View to identify particular links, and click on a page to highlight the details of that page on the list view. Wavefronts, hyperbolic trees, and the other graphical representations of Web site links and problems are ultimately a matter of taste. Displaying concentrated clusters of names, images, or icons is a hard user-inter- face problem wherever it comes up. Web- Analyzer costs \$299.

Linkbot Pro 4.0, from Tetranet Software (www.tetranetsoftware.com), is focused on fast, comprehensive link checking. The company claims it's no problem to check hundreds of thousands of pages without breaking down. The product also parses HTML and JavaScript code, flagging dozens of varieties of errors. In keeping with the concentrated capabilities of the product, the main display is a collapsible Explorer-like list of URLs that identifies error conditions. Because Linkbot Pro runs as a Windows NT service, it's easy to schedule regular site scans and reports. If you have one of those sites with hundreds of thousands of objects, the advanced filtering capabilities of Linkbot Pro can help make the task more manageable by limiting the scope. The single-user Linkbot Pro is \$295.

Along with link tests and missing file scans, LinkScan, from Electronic Software Publishing (www.elsop.com), checks name tags and references, which are the links within a single object that permit navigation within a file. As many of the other enterprise scale products do, LinkScan can be configured to access Web sites via a proxy server and to analyze dynamic pages, including those that use CGI, Active Server Pages, or database links. LinkScan has facilities for a single user to manage multiple sites, and for an individual site's responsibilities to be divided among multiple administrators. Error reports can be e-mailed to the individuals associated with a particular site or site subset. LinkScan can report redirected links, which lets the site manager anticipate future broken links and handle them before they become a problem. LinkScan costs \$750 per copy.

InfoLink Link Checker, from BiggByte Software (www.biggbyte.com), is a focused link-checking tool available as shareware for \$49.95. It offers proxy server support, the ability to configure multiple sites for regular checking, and an internal browser.

Xenu's ([www.snafu.de/~tilman/xenu link.html](http://www.snafu.de/~tilman/xenu%20link.html)) Link Sleuth supports Secure Sockets Layer (SSL) Web sites, detects and reports redirected URLs, and has an executable file size of less than 400Kbytes. It's available as a no-cost download.

Smart Site 2.0, produced by Big Picture Technologies (www.bigpic.com), has numerous configurable options that let users focus their scanning. It supports wild card characters (@* and ?) for defining URLs to scan. It lets users set the time-out period for connections. Along with catching broken links, missing pages, and HTML code errors, it

attempts to identify pages that can be expected to load slowly because they are oversized. The product includes a multilingual spell checker. (Correct spelling seems especially problematic on the Web, at least in English, so this feature may find it has a lot of work to do.) Smart Site will also analyze log files and report on traffic patterns. It is selling for \$189.95.

While most of the products I've discussed come from vendors who make, at the most, a handful of Web site oriented tools, there is a behemoth software company that takes an interest in the Web site content management area-Computer Associates (CA). MasterIT is part of CA's IT product line, which features a common Unicenter TNG core and some specialized management capabilities in each product. MasterIT's specialized capabilities include continuous Web link analysis, real-time traffic analysis and reporting, real-time response monitoring, and Web server monitoring. Unlike the other products that analyze links and log files after the fact, MasterIT reports events as they happen. With the Unicenter TNG framework built in, MasterIT can generate alarms and send them via fax, pager, or e-mail. The Unicenter connection also facilitates incorporating Web practices into an enterprise-wide set of policies and automating corrective actions. The Workgroup Edition of MasterIT normally sells for \$695, but is currently available free from the CAWeb site.

THE PROPHECY

As Web sites become more and more integrated into business operations, specialized tools for detecting problems with static Web pages are likely to decline in importance. However, integrated tools that ensure that the customers, partners, suppliers, and co-workers who get information and execute transactions over the Web are treated well will be as crucial as any other method for increasing customer satisfaction. The novelty of the Web will soon wear off (if it hasn't already), and customers will lose whatever inclination they now have to forgive bleeding-edge technological operations.

Steve Steinke, editor-in-chief, can be reached at ssteinke@mfi.com.

www.biggbYTE.com/index.html

Big Picture Technologies

Smart Site 2.0

www.bigpic.com

Computer Associates

MasterIT

www.cai.com

Electronic Software Publishing

LinkScan

www.elsop.com

InContext Systems

WebAnalyzer 2.01

www.incontext.com

Mercury Interactive

Astra Site Manager

www.merc-int.com

Tetranet Software

Linkbot Pro 4.0

www.tetranetsoftware.com

WebTrends

WebTrends Enterprise Suite

www.webtrends.com

Xenu

Link Sleuth

www.snafu.de/~tilman/xenulink.html

The Web Developer's Journal (http://Webdevelopersjournal.com/software/site_management_tools.html) is an online magazine that harbors reviews of numerous Web site management tools, including shareware and freeware programs.

One useful tool at the World Wide Web Consortium (W3C) is an up-to-date HTML validation reporter. You simply enter the universal resource identifier of a Web page and you can see if it complies with the standard. You can find it at <http://validator.w3.org>.

Web managers can detect broken links, clean up HTML code, trim down the size of files, and look at traffic patterns at great length without getting an objective conception of how a customer experiences the site, or of how well the site compares to other competing sites. WebCriteria (www.webcriteria.com), a Portland, OR-based company, provides these kinds of objective feedback to the owners and managers of Web sites.

The four criteria the company chooses to measure are load times, accessibility, freshness, and composition. Their methods focus on the site, leaving the impact of the communications network and the behavior of users out of the picture.

To measure load time, WebCriteria considers the number of elements on a page, the number of elements that repeat from one page to the next, the sum of the sizes of all the elements, the throughput available to the user, and the number of paths that are available to reach a page.

Accessibility is purely related to site navigation for WebCriteria-it has nothing to do with how easy the material is to understand or how transparent the logic of the site may be. The accessibility score is assessed by identifying the shortest path to each page on the site, taking into consideration the load times along the way and the number of choices that must be made.

WebCriteria measures freshness by examining the "last modified" date on the site's elements. Dynamically generated HTML pages won't have meaningful last- modified information, though most other objects will.

Composition for WebCriteria is a measure of the variety of experiences a browsing user will have at the site. Text, graphics, audio, video, applets, and files for download are the types of elements that establish that variety. However, the size of the files alone is an inadequate measure of the user's experience, so WebCriteria developed a "browsing behavior agent" to simulate the way users experience these different elements.

In order to create a SiteProfile Report, WebCriteria begins by sending a software " spider " to gather data about the site. The software makes use of the browsing behavior agent to scan the site's entire domain and capture all the relevant information-often as much as 100Mbytes of data. WebCriteria takes care to minimize the impact of the data gathering operation so it doesn't degrade site performance.

At WebCriteria's "factory" in Portland, the Web site data is used to build a model of the original Web site. Then the browsing behavior agent is turned loose on the model to establish measurements for download times, accessibility, freshness, and composition. The refined results of this model browsing are in turn used to generate a 16-page report, which includes three Web sites of your choice and one industry benchmark from a company whose business is similar to yours.

No doubt the state of the art for modeling and assessing Web site quality is in an early stage, but WebCriteria's efforts to provide objective, repeatable reports are a good first step.

COPYRIGHT 1999 Miller Freeman Inc.

COMPANY NAMES: WebTrends Corp.--Products; Mercury Interactive Corp.--

Products; InContext Systems Inc.--Products

GEOGRAPHIC CODES/NAMES: 1USA United States

DESCRIPTORS: Software multiproduct review; Web site management software

EVENT CODES/NAMES: 350 Product standards, safety, & recalls

PRODUCT/INDUSTRY NAMES: 7372682 (Internet Server Software)

NAICS CODES: 51121 Software Publishers

TRADE NAMES: WebTrends Enterprise Suite (Web site management software)--

Evaluation; Astra Site Manager (Web site management software)--Evaluation

; WebAnalyzer 2.1 (Web site management software)--Evaluation; WebTrends
(Web site management software)--Evaluation
FILE SEGMENT: CD File 275

9/9/9 (Item 9 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2006 The Gale Group. All rts. reserv.

02073899 SUPPLIER NUMBER: 19310966 (THIS IS THE FULL TEXT)
Publish dynamic applications on the Web. (using CGI to create dynamically
updated pages) (Internet/Web/Online Service Information) (Cover
Story) (Tutorial)
McClanahan, David R.
Databased Web Advisor, v15, n4, p60(1)
April, 1997
DOCUMENT TYPE: Cover Story Tutorial ISSN: 1090-6436 LANGUAGE:
English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 2702 LINE COUNT: 00236

ABSTRACT: The explosive growth of the Internet has made Netscape Navigator the most popular piece of software of all time and generated a demand for Web pages that distinguish themselves from the masses of useless information available. Common Gateway Interface (CGI) scripting lets Web developers add interactivity to pages and provide timely, flexible information. CGI is a specification for interfacing Web servers with external programs and helps overcome the limitations of HTML in creating dynamic pages. It defines an interface between Web servers and applications that may be written in any language; the CGI application accepts data from the server, performs processing and then returns data to the browser via the server. CGI uses a three-tier client/server architecture and can be invoked via a hypertext link or by clicking a button in a form. Information is passed to the application using environment variables and HTML code returned to the browser based on results of an action.

TEXT:

Go beyond HTML--use CGI to create dynamically updated web pages.

HIGHLIGHTS CGI Content types HTML

What is the single most popular piece of software ever created? More copies of this software product have been distributed than for Word, Excel, or Access. Hint: This product has only been available for a couple of years.

The software is Netscape Navigator. The answer tells us where the action is in our industry. The move to the Internet (and Intranets) is a recent development, but companies have been collecting data, building software, and using database systems for decades. So the question for many data professionals is how to tie current systems to the increasingly important, very hot technology of the Internet?

There are several solutions that let you push enterprise data to web pages. Do you know there is already a common, well-defined standard that provides the framework to allow you to link existing databases and/or database applications to your web pages? This is the Common Gateway Interface (CGI). In this article, I'll define CGI, and show several examples of how CGI can provide dynamic access to existing data or applications through interactive web pages.

Dynamic web pages

Today, most of what you see on the Web involves simple, static web pages. These are web pages that are fixed, and don't change (unless replaced or updated by the developer).

Client-server developers can also create dynamic, interactive web applications, which can tie a web page to new or existing back-end applications (or databases). The real power for the Web (at least for data professionals) begins with the discussion of dynamic web pages. Dynamic means that a web page can change (or be generated) interactively in response to the user's input through their browser.

Internet development has become an extension for our client-server

development, and developers are (or will be) required to implement **dynamic web pages** to interact with corporate data and business processes. Web development affects the requirements for our strategic planning and application development. However, you don't want to rewrite your existing systems just for the Internet. Instead, you can link your current systems (data or applications for this discussion) to your web sites.

There are several methods for implementing **dynamic web pages**. The first is with HTML forms. Forms were not in the original definition of HTML, but were added in version 2.0 to provide some interactive capability to web pages. The HTML form contains editable (input) fields which may include radio buttons, check boxes, push buttons, and lists, and always contains a Submit button. When the user clicks on the Submit button, the browser scans the input fields, formats the data into a message and sends it on to the server. The server examines the request to locate the destination (usually an application or a URL) and passes the data to the target.

Database and programmatic functionality

HTML is not a programming language and HTML forms are quite limited in capability. Another, more powerful alternative for creating **dynamic web pages** is to have the web server invoke back-end programs (applications or scripts) that execute in realtime. This method for implementing **dynamic web pages** lets you invoke a back-end program by embedding the program name in the URL that's sent to the web server.

This way, the client browser can pass data to the back-end application, and the application then executes and returns data to the browser (perhaps contained in a dynamically generated HTML page). A protocol has been defined to allow this backend communication between the web server and any external programs. That standard is called the Common Gateway Interface (CGI). The next section will explain this in detail.

Common Gateway Interface (CGI)

CGI defines an interface between web servers and external (CGI) applications for the purpose of performing some processing beyond the normal capability of the web server, thus increasing the power of web applications. This programming specification is also part of HTTP. In effect, CGI also defines an interface between web browsers, web servers, and back-end programs. CGI lets the user issue an action with the **browser** that **sends** a message to the web server that initiates some backend processing, and that may generate **dynamic web pages**. This will also allow you to connect your company database to the Web for interactive access.

It's important to understand that CGI is not actually a language, but only the standard definition for how the communication is handled between the server and external programs (running on the server or elsewhere in the system). CGI applications are usually called scripts. You can create CGI scripts in many languages including Perl, C, C++, or UNIX or NT shell scripts. CGI tells the developer how to send data to and from their applications.

The user can start a CGI application by entering information on a form or clicking on a link on a web page. The CGI application can perform any processing or database access and can also return a dynamic HTML page as a result. Figure 1 shows how CGI works. This model holds true for the other alternatives to CGI for creating **dynamic web pages** (which I will discuss in a future article).

(Figure 1 ILLUSTRATION OMITTED)

CGI applications

The CGI application (also called a CGI program or a CGI Script) accepts data from the server (usually input from the web page), performs its processing, and then returns data to the browser (via the server). The CGI application can (and often does) dynamically create a new web page. In this way, the contents of the web page are controlled by the user. For

example, a customer can search for certain items in your database. It would even be possible (though not necessarily a good idea) to store all of a web site's pages in a database, and to retrieve each page as requested by the browser.

CGI employs a three-tier client-server architecture; the first tier is the web browser client, the second tier is the HTML web server and the CGI applications, and the third tier contains the DBMS, TP monitors, and other resources required by the CGI applications. Once you have the ability to invoke a CGI application, you can do almost anything that you can currently do with client-server applications. With CGI, the web user can access new or existing applications and access (even update) a database. In this way, CGI allows you to extend your company's services to the Web.

Invoking a CGI application

HTML (a web page) allows the user to trigger a CGI command in several ways:

- * Via a hypertext link, which references a script to be executed, such as:

```
<A HREF=http://www.dmcclanahan.com/cgi-bin/getstatus.pli> Display  
system status</A>
```

- * By clicking on a Submit button on a HTML form. In this case, the web page will contain an ACTION tag:

```
<FORM ACTION=http://www.dmcclanahan.com/cgi-bin/ add_customeri>
```

As a result of one of these action, the web browser will issue a GET or POST command, send a URL, and usually data from the web page to the web server (more information on this follows). (The URL contains the reference to the CGI program.)

The difference between the GET and POST commands is how they package the information for the request. The GET command sends client data and a URL to the server in a combined string. Form data will be encoded and assigned to the QUERY_STRING environment variable. For example (this is a simplification):

```
http://www.dmcclanahan.com/cgi-bin/env.cgi?pg=q&what=  
Web&fmt=.&q=powerbuilder;Web
```

The POST command sends data to the server as a separate message. Form data is sent through the stain (standard input) for the system and the length of that data is sent in the CONTENT_LENGTH environment variable (this is considered the better method for programmatic use).

The server examines the URL and if it points to a CGI script (by searching the server's CGI script mapping) the server will execute the script. The URL often contain "cgi-bin" as path of the path for the URL. This is a common directory for CGI scripts.

Environment variables

CGI uses a set of environment variables to pass information to the CGI application. The server will initialize a set of environment variables which communicate information between the server and the CGI application and will pass any data on to the application. For example, the server will execute a program (env.cgi) in the server's cgi-bin directory as a result of this URL:

```
http://www.dmcclanahan.com/cgi-bin/env.cgi?pg=q&what=  
Web&fmt=.&q=powerbuilder;Web
```

In this example, the data follows the question mark, and each field is delimited by an ampersand. The env.cgi script just displays some of the CGI environment information. Listing 1 shows the result, which displays in the browser.

Listing 1: This example displays the values of some of the CGI environment variables.

```
AUTH_PASS:  
CONTENT_LENGTH: 0  
PATH_INFO: /cgi-bin/.nv.cgi  
PATH_TRANSLATED: C:\WINNT\system32\inetrv\cgi-bin\env.cgi  
QUERY_STRING: pg=q&what=Web&fmt=.&q=powerbuilder;Web
```

```

REMOTE_ADDR:      169.132.90.20
REMOTE_HOST:      169.132.90.220
REMOTE_USER:      dmcclanahan
REQUEST_METHOD:   GET
SCRIPT_NAME:      /cgi-bin/env.cgi
SERVER_NAME:      p90
SERVER_PORT:      80
SERVER_PROTOCOL:  HTTP/1.0
SERVER_SOFTWARE:  Microcoft-IIS/2.0

```

Listing 1 shows some of the CGI environment variables with values for this example. The QUERY_STRING is one of the most important variables. The HTML GET statement passes parameter data in the QUERY_STRING environment variable. In this example, it contains the string that follows the question mark in the URL. This contains a set of name/value pairs, each delimited by the ampersand (&). Table 1 shows the names and values in this example.

Table 1: CGI parameters for the QUERY_STRING environment variable.

Name	Value
pg	q
what	web
fmt	.
q	powerbuilder;web

The QUERY_STRING environment variable contains four name/value pairs.

As a result of the GET or POST command, the CGI application executes and returns its data as a HTML page (or some type of MIME data) to the web server. Finally the web server sends the result back to the browser that initiated the request. This may sound like a lot of work, but actually the CGI interface is fairly simple when you take into consideration the power it provides for creating dynamic web pages.

The web site manager must register the CGI application file extensions with the web server. Some of the options for executing scripts for web servers are shown in table 2.

Table 2: Once registered with the web server, the web site can execute various types of CGI scripts, like the ones shown here.

Extension	Interpreter
.bat .cmd	Cmd.exe
.idc	Httpodbc.dll
.pl	Perl.exe
.cgi	Perl.exe or PerlIIS.exe
.exe	System

Content type

Consider this link:

```
<A HREF=http://www.dmcclanahan.com/cgi-bin/getstatus.cgi> Display
status</A>
```

In this example, the getstatus.cgi script is a reference to a Perl script (you could use getstatus.pl). When executed, this script returns a simple text describing the up-time for the system as shown in listing 2.

Listing 2: This is the result of the invocation of getstatus.pl.

The browser displays this text on the screen.

Content-type: text/plain

```
The Development system is up and running since 10:14:22 AM
on 1/11/97
```

The server will return the file created by this script to the browser. The first line is a header that tells the browser the type of data

being returned. In this example, it's just a plain ASCII text, not an HTML page, and the browser will display the text accordingly.

Instead of returning a plain text file, you could return a dynamically created HTML page. In that case the previous example might display as shown in listing 3.

Listing 3: An alternative result of the invocation of display_status.pl.

Hello David McClanahan

The Development system is up and running since 10:14:22 AN
on 1/11/97

The HTML code that was returned to create this result is shown in listing 4.

Listing 4: The HTML code that was returned to the browser.

Content-type: text/html

```
<HTML><HEAD>
<TITLE>Display System Status</TITLE>
</HEAD><BODY>
<P>Hello David McClanahan
<P><P>The Development system is up and running since
10:14:22 AN on 1/11/97
</BODY></HTML>
```

The CGI application could generate this HTML code using C, Perl, or UNIX script (among others). To create the basic template in Perl, you would use the code in listing 5. Listing 6 shows an excerpt of the C version.

Listing 5: The Perl version of the HTML code shown in listing 4.

start.pl n creates a HTML page

```
print "Content-type: text/html\n\n";
print "<HTML><HEAD>\n";
print "<TITLE>i
0. Then print the actual title text
print i</TITLE>\n";
print "</HEAD><BODY>\n";
0 then insert whatever text you wish to send back
print "</BODY>\n</HTML>\n";
Listing 6: The C version (an excerpt) of the HTML code shown
in listing 4.
```

//start.c n creates a HTML page

```
printf ( "Content-type: text/html\n\n");
printf ("<HTML><HEAD>\n");
printf ("<TITLE>i);
//Then print the actual title text
printf (i%si, sTitle);
printf (i</TITLE>\n");
printf ("</HEAD><BODY>\n");
//then insert whatever text you wish to send back
printf (i%si, sText);
printf ("</BODY>\n</HTML>\n");
```

Notice that the CGI application simply writes its output to the system stdout (standard output). The web server picks this up and sends it back to the web browser .

Each language will access the environment variables in a different manner. The following Perl code shows how to print the value of all the environment variables:

```
while (($start,$end) = each(%ENV)) {  
    $line = $start . "=" . $end;  
    push(@envlist, $line);  
}  
for each $envvar (@srtenv) {  
    print $envvar, "\n";  
}
```

Since the data is encoded (as in the QUERY_STRING variable of listing 2), the CGI application must decode the parameters and must be able to handle both the GET and POST formats.

Other data types

In these examples, the CGI application has returned a plain text tile or an HTML page. The CGI application can return many other types of data including image and sound files. The data returned from the CGI application may contain a statement (a MIME Content-type statement header), which describes the type of data that it's returning. For a .GIF image, for example, you would use:

```
content-type:image/gif
```

The header is always followed by a blank line. (MIME stands for Multipurpose Internet Mail Extensions.)

It is also possible to send back a reference to another document as:

```
content-type: text/html Location:http://www.dmcclanahan.com/dmcpubl.htm
```

When the browser receives this (in a HTML page), the browser automatically moves to that page.

I have been describing non-parsed headers, which is a CGI method for handling return data by the web server where the data is simply returned to the browser. There's another method, called parsed headers, which requires interpretation by the server before returning data to the browser (a subject for another time).

Database access

Now it should be clear that CGI provides a method for executing any application you wish. This includes database applications that can access parameters and return any data as long as it is properly formatted using the CGI constraints. HTML includes a special tag, <ISINDEX>, which is often used to specify parameters for a database query. In this case, the query string is sent to the CGI application as command-line arguments. In a future article, I will focus on database access and provide examples for several platforms and DBMS systems.

Conclusion

CGI is a specification for interfacing web servers with external programs (in any language). This allows the creation of web pages with dynamic content; pages that can interactively access back-end resources such as applications and databases. CGI specifies the details on how to pass data to the scripts, and how the scripts return data to the server (and then back to the browser). The output of the CGI call usually includes a header that specifies the MIME content type of the return data.

CGI provides the method for accessing program code in most languages such as C or Perl. There is one caveat: Because CGI can invoke nearly any type of program (such as C programs), system security is an issue that you must consider. C programs can access the operating system, files, and other sensitive areas of your system; so you must put controls in place to ensure that security is enforced.

RELATED ARTICLE: Executive Summary

Web pages can provide timely information in an interactive environment through CGI. CGI isn't a programming language; it's a specification for interfacing web servers with external programs (in any language).

David R. McClanahan is an author, instructor, and consultant who

specializes in the design and development of client-server systems. He has been involved with every phase of development of many large client-server projects. Much of his work has been focused on Internet development and the creation of **dynamic web pages**. Visit his web site at <http://www.dmcclanahan.com/> for a list of publications and other information.

COPYRIGHT 1997 Advisor Publications Inc.

SPECIAL FEATURES: illustration; chart; program

DESCRIPTORS: Internet/Web Technology; Web Site/Web Page Development

FILE SEGMENT: CD File 275

9/9/14 (Item 14 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2006 The Gale Group. All rts. reserv.

01958605 SUPPLIER NUMBER: 18409456 (THIS IS THE FULL TEXT)
Put dynamic data in your Web pages. (includes related articles on linking
databases to the Web and JYACC's JAM/Web) (Internet/Web/Online Service
Information)
Youngworth, Paul
Data Based Advisor, v14, n7, p62(8)
July, 1996
ISSN: 0740-5200 LANGUAGE: English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 3100 LINE COUNT: 00251

ABSTRACT: Internet-aware databases, Web server utilities for databases, programming interfaces and Internet-specific scripting languages can save Web developers from the tedium of writing Perl CGI scripts to connect databases to the Web. Oracle's Web Server features standard PL/SQL syntax and is a good choice for use with Oracle databases. Informix' Illustra is a similar product that employs an Illustra server separate from the Web server. Sybase's web.sql takes a different approach, offering a database interface that reduces coding necessities. Informix also offers interface freeware products on its Web site. For applications in which data is stored in a variety of formats, a database gateway may be the best choice. Information Builders' EDA Web Connect is an API in development that connects over 60 data formats and databases to CGI scripts. Powersoft's Optima++ is a C++-based tool for developing Web-enabled databases. Microsoft's Visual Basic Script and Oracle's Oracle BASIC are Internet-specific scripting languages.

TEXT:

This message posted in an Internet newsgroup had a feel of desperation: "I designed a demo web page that displays 10 products," the note said. "It allows the user to click on a hyperlink for each product to bring up more information. Now my boss wants me to build a web page that lets potential customers drill down through our entire product line! I'm going crazy trying to write CGI scripts that manipulate all this data. Does anyone know of an easier way?"

Does this situation sound familiar? The note above is a paraphrase of numerous messages now showing up online. Companies are no longer satisfied with presenting a few static web pages to potential customers. They want customers to interact with live data, fresh from corporate databases, and they want you to provide that capability. Coding a ton of CGI scripts to retrieve, manipulate, and format data just won't cut it anymore.

Ross Morgan, project leader of Internet applications for Investors Group says developers can't afford to write the "reams of Perl or C code that it takes to get web pages and databases to talk to each other." Morgan has been programming Internet applications for four years, and has dedicated the last two years to web page development. He says the explosion in database-driven web pages has changed the way developers approach the Internet. "Developers need tools that take the pain out of developing CGI applications," he says.

To address this need, Oracle, Sybase, Informix, Information Builders, and Microsoft have all recently introduced products that make it easier for us to link web pages to corporate databases.

Part one of this series ("The Rush to the Net," May 1996) gave an overview of how each of these companies is approaching web page development. In this article, I look at what each approach means to you, the developer, and examine the products they offer in terms of:

1. The advantages and disadvantages of each approach. This includes the advantages and disadvantages of databases that directly produce HTML,

and of using a proprietary browser.

2. The way you create applications using these products. I show some sample code and screen shots of applications and development environments.

3. The impact of a company's full product line. Do any of the products support Java? .OCXs? How open is the database access?

The best way to categorize these products is to look at where they fall in the process of communication between the browser and the back-end database.

Web-enabled databases

One way to make things easier for developers is to reduce the amount of CGI scripting by having the database itself create HTML documents. This is the approach taken by Oracle's Web Server and Informix's newly acquired Illustra. Oracle says this approach, used by its Web Server, is the best way to deliver personalized web pages. Figure 1, taken from Oracle's web site at <http://www.oracle.com>, highlights this strategy.

Web Server stores the data and the program that turns the data into HTML in the database. The browser calls a PL/SQL stored procedure that retrieves the data and formats it as an HTML document. In effect, the database encapsulates the data and the method of delivering that data.

Figures 2 and 3 show an Oracle Web Server application in action. Figure 3 shows the results of the query in figure 2 as they're returned to the browser. This sample application was developed by Thomas Dunbar, and is displayed on Virginia Tech's web site at <http://gserver.grads.vt.edu>.

The application is developed using standard PL/SQL syntax to retrieve the data. It also uses new PL/SQL extensions that write the output in HTML syntax. Some of the extensions are:

- * http.htitle
- * http.formopen
- * http.print
- * http.bodyclose

You mix these commands in with traditional PL/SQL code to create dynamic HTML code.

Illustra's Web DataBlade Module uses a similar approach to deliver a web page to a browser. Figure 4 shows the process Illustra uses to build a web page.

The Illustra server stores its Application Pages in the database, along with the data. An Application Page is the web page HTML code with additional standard SGML tags, and special database-oriented markup tags called Web DataBlade tags. These additional tags let developers embed SQL calls in the application Page and apply formatting to result sets. You develop Illustra web pages by mixing the standard and special tags in a document with the SQL calls that return the desired data. Application Pages are stored as rows in the Illustra database.

When the Illustra server receives a request from a web server, it retrieves the appropriate Application Page from the database. An Illustra module called WebExplode parses the Application Page, executes any embedded SQL, and formats the results. An HTML document is returned to the web server. All web page processing is done on the Illustra server itself. You can store audio, a wide variety of graphic formats, and the traditional data types in the Illustra database along with Application Pages. This lets you organize web pages and associated resources in the same data store.

Illustra maintains a web page containing hyperlinks to web sites built using the product (figure 5).

One obvious advantage of Web-enabled databases is a familiar development environment. Oracle's Web Server is an attractive route to the Web, if you already have your enterprise data in Oracle databases.

Kyle Brown runs a consulting company called Project Solutions. He's currently developing an intranet for a large company that uses Oracle Financials, and stores order data in Oracle databases. He says, "Oracle Web Server becomes a leading candidate for that situation. You have to take a look at it."

Web-enabled databases allow you to bypass a lot of CGI programming, and do that processing right on the server. They also simplify the organization of web page contents because they store the web page logic, graphics, and other resources in the database.

However, there's also a down-side to Web-enabled databases: They're a proprietary solution. Brown's largest client has both Oracle and Sybase databases. Oracle Web Server only solves the Oracle side of the Web-to-database problem.

Investor Group's Ross says he's looking at Oracle Web Server, but has reservations. "I don't like the fact that Oracle Web Server only uses PL/SQL to create the HTML," he says. "It still takes a lot of coding to get attractive web pages."

Programming interfaces

Providing database interface software that reduces some of the coding is another way to simplify database access from web pages. Sybase's web.sql takes this approach. This programming interface sits between the web server and a Sybase database. The diagram in figure 6 shows web.sql's role in the process. Web.sql uses a file format called HyperText Sybase, or HTS. An HTS file is an HTML document with added <SYB> tags. You write the HTML code as usual, except you must put either Transact-SQL statements or Perl code between <SYB> tags. This code calls the Sybase databases and returns the result set, which is placed in the appropriate area in the final HTML document.

Here is some web.sql code that, when given a department name, returns a list of employees:

```
<H1>Employees</H1> Employees in the $dept department:<P> <SYB
TYPE=SQL>
```

```
select lastname, firstname, phone
from emp
```

```
where dept upper("$dept")
```

```
order by lastname </SYB> <!-- more HTML coding ..... -->
```

web.sql still requires Transact-SQL and Perl programming, so how does it reduce the amount of code required? Al Huntley, a Sybase database administrator for Lockheed Martin Energy Systems, says the sample templates that come with web.sql can make a developer really productive.

"web.sql is really easy to use," Huntley says. "I've been able to produce 30 to 40 web pages in a week by copying the sample programs and pasting in my own SQL statements. You can crank things out quickly."

Informix offers two web page interface freeware products: Informix-ESQL/C CGI Interface Kit and Informix-4GL CGI Interface Kit. These products are libraries of functions that developers can use to access Informix data, and write the output to an HTML document. Informix's web site at <http://www.informix.com> has several demos that show the functionality of these products. The web pages let you run a demo and then look at the executed code.

Figure 7 shows an Informix demo application. In this demo, you key in a customer name or company, then click a command button to retrieve related data from an Informix database. Figure 8 shows an example of the code you can view at the Informix web site.

Programming interfaces, like web.sql and the Informix interface kits, can cut down on the coding needed to tie web pages to databases. These interfaces can be especially useful to Web developers with experience writing CGI programs, who are looking for templates that reduce the amount of repetitive coding.

However, these products are proprietary, and you still need to write code. The products can be valuable if your application is like one of the sample apps included in the product. If not, you'll have to do a lot of programming.

Database gateways

If your data is stored in a variety of formats, Web-enabled databases and programming interfaces only solve part of the problem. In this case, a

product like Information Builder's EDA/SQL may be the best solution.

John Deutsch, director of product marketing at Information Builders, says the company will soon release a product called EDASWeb Connect is an API that links CGI scripts to more than 60 databases and data formats. "EDASWeb Connect acts as a bridge between the Internet and a corporation's data," Deutsch says. "You can use one API to access a heterogeneous mix of data sources."

Taking the EDA/SQL approach requires more coding than other situations, but might be the only viable alternative in certain situations.

Web-enabled application development software

Another way to build web pages that access data from multiple sources is to use a database-independent application development tool that can create HTML as its output. Powersoft's newly announced Optima++ is a component-based development tool that uses C++ as its language. Developers use Optima++ to develop content and applications for web servers.

Optima++ is a GUI development environment. Figure 9 shows a sample screenshot from Powersoft's web site, <http://www.powersoft.com>. Optima++ includes the data-aware technology in PowerBuilder and InfoMaker, such as the DataWindow control and the Pipeline control. Optima++ gives developers point-and-click access to all major databases, which simplifies the chore of accessing and formatting data.

Powersoft says Optima++ will create executable content for web pages that can be delivered as Java applets or Microsoft Sweeper controls.

Deutsch said a forthcoming release of IBI's Focus Report Writer will allow developers to designate HTML as the output for Focus reports. "The developer can use stylesheets to indicate which columns are to be hyperlinked," he says. "This will let you build drill-down web page applications."

According to Deutsch, Information Builders is also planning to release a multi-tier development tool called Cactus. Developers will use Cactus to develop applications for web servers, and to interact with Java Applets embedded in web pages.

Optima++ allows you to generate Web applications that draw data from different databases. This type of product has the potential to greatly simplify the development of data-intensive web pages.

For now, the biggest disadvantage of this type of software is that no one really knows much about it. Products like the Oracle Web Server or Sybase's web.sql may be proprietary, but at least developers know what to expect. Eventually, the new generation of Web development tools may simplify Internet and intranet development, but at this point it's yet another technology to learn.

Scripting languages

Microsoft and Oracle have announced Internet scripting languages. Visual Basic Script from Microsoft and Oracle BASIC from Oracle are intended to handle web page data processing on both the server and the client.

Visual Basic Script is a subset of Visual Basic, and Oracle BASIC is clearly modeled after Visual Basic. Both products will be familiar to the legions of Visual Basic programmers. This advantage will open Internet/intranet programming to a wider group of developers than programming solutions based on C++.

Microsoft has announced plans to deploy Visual Basic Script across multiple platforms. Initially, however, it only will be available on Windows NT and Windows 95. Java, on the other hand, is targeted for all major platforms simultaneously.

Scripting languages that are not available across platforms will be more useful for intranet rather than Internet applications. Project Solutions' Brown develops both intranet and Internet applications for clients. "I'm not going to buy Intel processors for remote users just so they can link to my Internet site," he says. "But it's a nice technology for controlled environments."

Investor Group's Ross says the different approaches to scripting languages could hurt Internet development in the short term. "There will be Microsoft developers in one camp and cross-platform Java developers in the other. It could turn into a battle to see who wins the hearts and minds of the industry."

Other pieces of the puzzle

Both Microsoft and Oracle have announced web browsers. Microsoft is introducing the Internet Explorer and Oracle has released the PowerBrowser. Both products exploit the companies' scripting language for client browser processing. Microsoft has acquired a web page authoring tool called Front Page.

If you're developing an intranet, Microsoft's and Oracle's strategy of having products on both the server and browser side of web page processing could be an advantage. If you develop for the Internet, you usually won't be able to take advantage of this two-sided technology.

The above matrix summarizes the product offerings from Oracle, Sybase, Informix, Information Builders, and Microsoft.

The products I discussed in this article just scratch the surface of the many Web-to-database products available. One web site (see side bar) lists 45 products for UNIX platforms, 23 for Intel-based servers, and six for the Mac environment.

You can link your web pages to corporate data without writing a ton of code. As developers gain experience with these products, **dynamic web pages** will become the norm and not the exception.

Web developer Kyle Brown summed it up best when he said the new generation of Internet development software "just opens up all kinds of possibilities."

(TABULAR DATA OMITTED)

RELATED ARTICLE: Web Pages and Databases: Where to Go for Information

If you want more information on how to link web pages to corporate databases, visit the World Wide Web. There are a number of web sites with useful information on this topic.

The software companies mentioned in this article are using their web sites to make it easy for you to evaluate, and even experiment, with their software. Many of the web sites offer:

- * Online white pages explaining the technology, often with hyperlinks to more detailed information.
- * Live demos you can run, along with a way to view the code behind the demo.
- * Hyperlinks to customer web pages built with the software company's products.

- * A way to download free or evaluation copies of their software.

Here's a list of web sites and what to look for:

COMPANY WEB SITE NOTEWORTHY

Sybase <http://www.sybase.com> Free copy of web.sql

Oracle <http://www.oracle.com> Trial Web Server

Informix <http://www.informix.com> Free Interface Kits

Information Builders <http://www.ibi.com> Web3270

Powersoft <http://www.powersoft.com> Optima++

screen shots Illustra <http://www.illustra.com> Hyperlinks to

customers Microsoft <http://www.microsoft.com> Free Internet

Assistant

Numerous other web sites contain information on linking web pages to databases. Two of these stand out from the crowd. One, maintained by Web page developer Jeff Rowe of COMSO, Inc., includes extensive information located at http://cscaun1.larc.nasa.gov/beowulf/db/web_access.html. This site contains a number of white papers that serve as tutorials on the concepts behind accessing databases from web pages. It also offers a comprehensive list of the web/database gateway products, with a paragraph about each. The second is at <http://gserver.grads.vt.edu/rindex.html>, and

is run by the graduate school at Virginia Tech. The site is dedicated to information on linking web pages to Oracle databases.

These web pages show several examples of applications and code. The applications start simply and gradually get more complex, ending with applications that take advantage of the frame technology in Netscape 2.0 and Java-based applications. There's a demo app that uses the Java Spreadsheet class to embed a four-function calculator in a web page.

Both web sites have numerous hyperlinks to other sources of information about linking web pages and databases.

RELATED ARTICLE: Web Pages and Databases: Another Approach to the Web

While Powersoft is introducing a completely new product for Internet development, JYACC, Inc. is taking a different approach. JYACC is adding Internet extensions to its existing multi-tier development software, JAM.

JAM/Web is the same development environment as JAM, with the additional capabilities needed to deploy applications on the Internet. JAM/Web developers can design forms, and deploy them on a web server. When a browser calls for a form, JAM/Web dynamically converts it into HTML and sends it to the end-user for processing. Fields from the form are sent back to JAM/Web for processing.

Developers can take components of existing JAM applications, and use them in Internet and intranet settings. All the elements of a JAM application are available for web page processing: forms, table browsing, reports.

JAM/Web lets JAM developers create Internet applications without having to learn a new tool or write interface routines. And JAM/Web will offer the same multi-platform, database independent development environment as the base product.

Kyle Brown of Project Solutions says his largest client already has many applications written in JAM. JAM/Web is a quick way to incorporate the data from those applications in Internet/intranet systems.

For more information on JAM/Web: JYACC, Inc. <http://www.jyacc.com>
(212) 267-7722

COPYRIGHT 1996 Data Based Solutions Inc.

SPECIAL FEATURES: illustration; table; chart

COMPANY NAMES: Oracle Corp.--Products; Informix Corp.--Products;

Information Builders Inc.--Product development; Powersoft Corp.--Products
; Sybase Inc.--Products

DESCRIPTORS: Internet/Web Technology; Database; Internet/Web Server
Software; DBMS; DBMS Utility; Application Development Software; Database
Middleware

SIC CODES: 7372 Prepackaged software

TICKER SYMBOLS: ORCL; IFMX; PWRs; SYBS

TRADE NAMES: Oracle WebServer (Internet/Web server software)--Design and
construction; Illustra (DBMS)--Design and construction; EDA Web Connect
(Database middleware)--Product development; Optima++ (Application
development software)--Design and construction; Sybase web.sql (Database
middleware)--Design and construction

FILE SEGMENT: CD File 275

9/9/49 (Item 3 from file: 647)
DIALOG(R)File 647:CMP Computer Fulltext
(c) 2006 CMP Media, LLC. All rts. reserv.

01173856 CMP ACCESSION NUMBER: WIN19981001S0026
Drumbeat Beats Other Web-Dev Tools (Software)
Baraka Dorsey
WINDOWS MAGAZINE, 1998 , n 910, PG104
PUBLICATION DATE: 981001
JOURNAL CODE: WIN LANGUAGE: English
RECORD TYPE: Fulltext
SECTION HEADING: Reviews
WORD COUNT: 753
TEXT:

I've been jilted by Web-development tools so often I'd begun to believe I would never find the right one. Sure, they all sweet talked me with promises of being easy to use, affordable and "the only complete, turn-key solution for all your development needs." But when I put them to the test, every one fell short. So when I saw Drumbeat 2.0 from Elemental Software, I was prepared for the worst. But what I discovered was the best all-around Web development software I've seen.

The first thing you need to know about Drumbeat is that it isn't a high-end tool like Microsoft's Visual InterDev. And if you're only concerned about producing static Web sites, you're probably better off with a tool such as HomeSite or FrontPage. But if you're a nonprogrammer interested in moving from static Web sites to building dynamic, database-driven sites, this is the tool for you.

Smart, powerful features

Drumbeat's ready-made templates and site-management and publishing features range from good to great. Its best features are the DataForm Wizard, SmartPages, SmartElements and Point-and-Click Interactions. Using the DataForm Wizard, I was able to create Active Server Pages (ASP) and complex SQL server queries in a matter of minutes-and I have no experience doing either. The wizard sets up all the code. All you need is the data.

Normally, creating a DHTML-rich Web site is impractical at best, requiring several versions of a page to ensure cross-browser compatibility. Drumbeat allowed me to create a Web site that includes DHTML multimedia effects, data binding and other client-side interactivity; in addition, its SmartPage feature created a version of the pages for various browser groups and automatically generated the client-side, CGI and ASP redirect pages to ensure that site visitors would see pages compatible with their browsers. The browser groups that Drumbeat supports range from Generic (Internet Explorer 3.0, Navigator 2.0, CyberDog and AOL)-which features JavaScript, Java and tables-to IE 4.0, which features scriptlets, IE 4.0's multimedia effects, ActiveX controls and data binding.

Drumbeat's SmartElements are components you can drag and drop anywhere on your pages to represent DHTML, client or server-side JavaScript, ActiveX, Java, CGI, COM or scriptlets. Behind the scenes, each SmartElement generates code that works properly for the specific browser the visitor is using. And, yes, you can create customized SmartElements with the Drumbeat SDK. Point-and-Click Interactions build impressive behaviors and interactions without coding.

For any combination of objects on your page, Drumbeat creates a plain-English, pop-up list of all the valid interactions available to those objects and the target browser. And you're not limited to only one interaction per object group. The program ships with a number of Interactions (DHTML filters, transition effects, media controllers and players, form validations and database Interactions), and lets you create your own customized Interactions as well. The supplied Interactions are more than enough for most people-it's just a bonus to have the option to create your own.

Behind the scenes ...

I was a bit disappointed that I couldn't view and edit my test page's source code from within the program, but when I took a quick look at the page's source code using HomeSite, I found that the HTML and JavaScript generated by Drumbeat were better than anything I could do by hand. I was also surprised to see that Drumbeat altered the source code of any pages imported into the program. However, if you tell the program to simply include the file in your Web site for management purposes, you'll be able to view, link to and publish the file without the program modifying it.

Drumbeat's documentation is the best I've ever read. It's written in plain English, so you'll have no problem finding the information you need. This is a good thing, because the user interface is so unique you'll want to run through the tutorial to get familiar. Once you finish the tutorial, you'll find Drumbeat easy and-dare I say it- fun to use.

Be warned, however, that all of that fun and simplicity isn't cheap. At around \$700, Drumbeat 2.0 is probably the highest priced development tool in its class. Luckily, it's in a class by itself, which is why we've placed it on our WinList.

-Quick View-

Drumbeat 2.0

Bottom Line: Best entry-level, database-driven Web-site development tool

Price: \$699

Platforms: 95, NT

Pros: Powerful, easy-to-use features; cross-browser publishing; professional results

Cons: Price is high for entry-level tool; can't view or edit source code

Elemental Software, 877-DRUMBEAT, 760-931-7171. Winfo #744

Copyright 1998 CMP Media Inc.

COMPANY NAMES (DIALOG GENERATED): Bottom Line ; Elemental Software ; Microsoft

9/9/57 (Item 4 from file: 148)
DIALOG(R)File 148:Gale Group Trade & Industry DB
(c)2006 The Gale Group. All rts. reserv.

09298072 SUPPLIER NUMBER: 19081469 (THIS IS THE FULL TEXT)
Information management: Toolkit. (Intranet - the inside Internet)
Socka, George
CMA - the Management Accounting Magazine, v70, n7, p25(2)
Sep, 1996
ISSN: 0831-3881 LANGUAGE: English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 1182 LINE COUNT: 00098

ABSTRACT: Intranets, the internal information publishing networks, can be extremely valuable in serving a limited audience and can be created with relatively small expense and readily available components. A Web server acting computer is needed to transmit information, a computer with Web-browsing programs is needed for viewing, and a network to interconnect the computers are sufficient components for meeting Wide World Web and Internet standards.

TEXT:

Two of the most widely discussed information technology topics recently have been business use of the Internet and the World Wide Web. Two significant factors make them so alluring. The Internet architecture allows any computer that is connected to the network to communicate with any other computer, regardless of who manufactured it, and regardless of what operating system it uses. This inter-connection capability is provided by a communication protocol named TCP/IP, which by design (and as a result of the economic pressure exerted by the U.S. Government purchasing standards) is understood by virtually every currently manufactured computer. The World Wide Web subset of the Internet is accessed by a Web browser on the user's computer, a program that can display "pages" of information published by a corresponding Web server. Any Web server can talk to any Web browser. Web browsers and servers are available for virtually every brand of computer.

This ability to interconnect computers, and to universally publish and read information, makes Internet and World Wide Web technology an ideal toolkit for publishing business information to a wide audience without the need to invest significantly in infrastructure, especially at the user's end.

There are many reasons to publish information on the Internet to the whole world. There are probably many more reasons to publish information internally. An Intranet is an internal network that uses the Internet and World Wide Web standards for a limited audience.

Creating an Intranet requires three components:

- * A computer that acts as a Web server;
- * Computers for the target audience with Web browser software;
- * A network to interconnect these computers.

The server

Web servers can be UNIX computers, from PCs with SCO UNIX or the free Linux version of UNIX to IBM RS/6000 computers with AIX, Microsoft Windows NT servers, IBM AS/400s and even IBM mainframes. Web serving can co-exist with other programs, or can be done by a computer dedicated to the task. The amount of load placed on the server will be dependent on the number of users who will access Web pages at a time, as well as the complexity of the process used to create the information being published.

The server computer needs Web server software, available from IBM, Microsoft, Netscape, Sun, Oracle and others. Microsoft's NT Web server software is free, as is Novell's. Most of the others sell Web server software as ancillary to their main product lines, and have marketing deals from time to time. In addition to the Web serving software itself, a tool is required to create or reformat the information to be published into the

hyper text markup language (HTML) that the browsers understand. Microsoft Word or Corel WordPerfect can be used to create " static " Web pages , and more complex pages which include up-to-the-minute information from databases such as inventory, can be created using authoring tools from Oracle or IBM.

Intranet Web servers, based on NT or Sun UNIX computers can be purchased as complete, turnkey systems to eliminate the configuration effort and minimize the disruption of other business systems.

The network

The network connecting computers can be anything from a local area network to a global, wide-area network, as long as the servers and users can communicate. The TCP/IP network protocol is a standard because of the Internet heritage, but Novell has a product that allows an Intranet to be created within a Netware IPX network. Most networks with UNIX computers in them already use TCP/IP; and adding TCP/IP to other types of networks is only modestly complex. Windows and DOS computers will require additional software, either free from Microsoft's Internet site (www.microsoft.com) or from independent vendors such as Hummingbird. Windows 95 has built-in TCP/IP, as do newer Apple computers. TCP/IP identifies each computer with a cryptic Internet style "address," and a significant amount of planning is required to assign addresses to a network covering multiple geographic sites.

While most Internet connections use slow speed modems, internal local area networks will offer speeds many times faster.

The audience

Publishing information is useless unless it can be read. The end user's computer requirements are modest once the network connection has been set up. A Web browser program is all that's needed. Even Word and WordPerfect have options that allow them to find and read HTML documents published by a Web server, but again, the fact that most Web browsers are free makes them ideal for widespread implementation. An issue to note is that while all Web servers and browsers in theory adhere to the same set of standards, Netscape and Microsoft servers provide enhancements to "standard" HTML that all browsers may not be capable of displaying. Planning and coordination will prevent embarrassing situations.

The business case

The fact that technology exists is not justification for its implementation. While other technologies can be used to publish and read information, Intranets have some advantages:

- * The server, browser and network technology is available for virtually every brand of computer system, and there is no need for all of the hardware, software and network components to come from the same vendor.

- * The server and browser functions can be added to computers already in place for other purposes, and the TCP/IP network protocol can be added to existing networks that use other protocols.

- * The browsers require little training to use, and creation of Web pages can be done with little more than a word processor.

- * The Web page HTML format, with its namesake hyper text linking capability that allows the user to jump from one document to another is particularly suitable for information such as office directories, procedures manuals, catalogues, press releases and marketing information.

Like most computer technology implementation projects, however, there are a number of factors to consider and plan for. These include:

- * Adding the unstructured and difficult-to-predict-load of Web page serving to an existing server may overload it to the point that other functions may become unusable.

- * Adding TCP/IP to an existing network protocol may render it inoperable, and may require replacement of components such as bridges and routers .

- * Adding browser software to anything except the latest Windows 95, OS/2 or Apple system may require extensive trial and error testing,

especially if the computers already have Internet access software installed on them.

- * Creating **dynamic Web pages** with database information or Java "applet" mini-programs is a real programming job, and an ongoing information systems management job to keep up to date, and working.

- * Implementing security to restrict certain information to certain users requires planning and ongoing management, and additional "firewall" hardware if the network is also connected to the Internet in any way.

George Socka, CMA, CMC, is a Toronto-based management consultant specializing in information systems strategic planning and management.

COPYRIGHT 1996 Society of Management Accountants of Canada
(Canada)

INDUSTRY CODES/NAMES: BANK Banking, Finance and Accounting; BUS
Business, General; BUSN Any type of business

DESCRIPTORS: Intranets--Management

PRODUCT/INDUSTRY NAMES: 4811500 (Specialized Telecommunication Services)

SIC CODES: 4822 Telegraph & other communications

FILE SEGMENT: MC File 75

9/9/58 (Item 5 from file: 148)
DIALOG(R)File 148:Gale Group Trade & Industry DB
(c)2006 The Gale Group. All rts. reserv.

09195683 SUPPLIER NUMBER: 18983858 (THIS IS THE FULL TEXT)
An a-to-z Web server suite. (Microsoft Commercial Internet Systems Web
server software) (Software Review) (Evaluation)
Schultz, Keith
CommunicationsWeek, n642, p41(2)
Dec 16, 1996
DOCUMENT TYPE: Evaluation ISSN: 0746-8121 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 2623 LINE COUNT: 00204

ABSTRACT: Microsoft's Commercial Internet Systems (MCIS) Internet/Web server software package offers Web site developers and Internet service providers (ISP) a bundled series of components for Web site creation and maintenance. Formerly named Normandy, MCIS includes the Microsoft modules Chat Server, Content Replications System, Merchant Server, Internet Locator Server, Personalization System, SMTP/POP3 Mail Server and Internet News Server. MCIS simplifies the creation commerce-enabled Web sites by including all components in a single package, reducing interoperability concerns with products from multiple vendors. However, users must be sure they have an Open Database Connectivity-compliant database and a server with at least a 133MHz Pentium and 64MB of RAM.

TEXT:

Just as Microsoft did a couple of years ago by bundling Word, Excel and PowerPoint into Microsoft Office, Internet service providers (ISPs) and intranet developers will be able to purchase a suite of commercial-grade Internet applications called Microsoft Commercial Internet Systems (MCIS). Consisting of six separate programs, formerly known by the code name Normandy, MCIS is tailored to provide an ISP with all the building blocks needed to set up a stable, scalable and dynamic Internet site capable of handling thousands of hits per day. Just be sure to base your commercial site on high-performance server hardware. A realistic hardware minimum for MCIS is at least a 133-megahertz Pentium with 64 megabytes of RAM.

Included in MCIS are Microsoft Internet Locator Server, Microsoft Chat Server, Microsoft Personalization System, Microsoft Content Replication System, Microsoft Internet News Server, Microsoft Merchant Server and an SMTP/POP3 Mail Server. As of press time, all the servers covered here are in beta 2 or "gold code" versions and pricing had not been set.

Microsoft will make some of the individual servers available independent of MCIS, but the News Server and Mail Server will only be available in MCIS. We'll take a look at each application in the package with the exception of the Internet Locator Server (ILS) and the SMTP Mail Server. A beta of ILS was not available at press time because of a small bug discovered the day the software was to be released, and the SMTP Mail Server is not going to be available until MCIS ships in the first quarter of 1997.

Microsoft has eliminated many of the headaches of setting up a commerce-oriented Internet site. You no longer have to use different products from different vendors and hope they work together, or stay up late hacking the latest Linux kernel to get your Network News Transfer Protocol (NNTP) server working. All the applications in MCIS are tightly integrated into Microsoft's Windows NT Server and Internet Information Server (IIS). Each system provides counters for Performance Monitor so that you can monitor the health of your site. In fact, you can even spread a system's processes across multiple NT servers.

Merchant Server, for example, has a number of processes specific to it. You could have the database engine on one server, the storefront on a second and the transaction processing engine on a third. This gives you the

most effective way to balance the CPU load at your site without requiring an expensive investment in huge, multiprocessor server machines.

Installation of all the MCIS server suite components is straightforward, as for most Microsoft products. For best results, make sure you have Windows NT Server 4.0 installed and are running the beta of IIS 3.0 (IIS 3.0 will come with ActiveX Server pages, which the Personalization System needs). IIS 3.0 should be available at the same time MCIS is released, and you can find the latest versions of MCIS's individual servers and IIS 3.0 beta on the Microsoft Web site (www.microsoft.com).

The only other real "gotcha" to watch out for with this suite is that you must have an Open Database Connectivity (ODBC)-compliant database, such as Microsoft SQL Server or Access, available for Merchant Server. If you don't, Merchant Server will install just fine, but it won't function. Microsoft recommends that the minimum server hardware be at least a P133 with 64 megabytes of RAM. Unfortunately, our test server was an IBM PC Server 320 P100 with only 32 megabytes of RAM, and even under a light load, at times the system would slow to a crawl. Although the performance from the client accessing the server never showed any performance degradation, the server itself would respond slowly to commands and even mouse movements.

Microsoft Chat Server

One of the best ways to kill a few hours on the Internet is to join in on the world's largest party line-Internet Relay Chat (IRC). For site administrators interested in offering this service, Microsoft's Chat Server (MCS) is one of the many IRC server packages that work with Windows NT. It also works with any IRC client, but has other bonuses as well.

As the site administrator, you can configure every aspect of MCS, such as the maximum number of connections and those to which users do not have access. All this is easily managed from MCS's property sheet in NT's Internet Service Manager. You also can modify MCS's software settings via your workstation's Web browser and an included HTML-based configuration interface.

A feature you won't find in any other chat server is support for the Microsoft Internet Chat (MIC) protocol. In fact, you must use Microsoft's chat client to utilize MIC. What makes MIC special, according to Microsoft, is that its data packet is smaller than traditional IRC packets, making it potentially more efficient over slower dial-up lines. We weren't able to test this formally, but performance to our test client was pretty good.

You also can initiate private chats and even chat queries over MIC. One of the best features, however, is its ability to easily pass along data-based messages for Internet game playing, and you can even configure MCS so it is in "game-only" mode. MCS also can connect to other MCS servers from a larger network.

If you have ever been to www.msnbc.com and configured the site to offer only the information you're interested in, you've already seen Personalization System in action. Site administrators are finding that Web users are no longer satisfied by solely static Web sites where they see the same page each time they log on. The next evolution will involve dynamic Web sites that show users' individually created Web pages based on criteria they have previously selected. The Microsoft Personalization System is the engine that keeps track of your users and generates the correct Web page content for them.

How the Personalization System Works

The way Personalization System works is a user fills in a form that describes him or herself. The system takes this information and dynamically generates a Web page that displays the content the user requested. In the case of MSNBC, you can filter the news you want based on category and keyword. Users are identified by an entry made to the browser's "cookie" file. This globally unique ID (GUID) allows for unobtrusive user identification when the user returns to the site. Don't worry. Microsoft says the 32-bit GUID is the only thing read from the cookie. We found the

Personalization System to be one of the best features of the MCIS package.

This server requires at least the IIS 3.0 beta or later because of the liberal use of ActiveX server pages. This also is one of the most potentially hardware-greedy of the new MCIS packages. Be sure to run this one with plenty of hardware juice, because system demands can be heavy.

Microsoft Content Replication System

We found Microsoft Content Replication System (CRS) to be one of the more useful Internet applications in recent memory. With the explosive growth of the Internet, companies have found that they need a way to keep multiple sites' Web information better synchronized. Until now, you had to use scripts or clunky batch files to accomplish these tasks. CRS is a great means of updating or synchronizing all your sites easily and automatically across the Internet or the local intranet.

Using CRS, you have many ways to distribute your Web content to other sites. You can use standard replication to move files that have been changed or added since the last replication, or you can use incremental replication to only copy the files that have changed. Transaction-based replication is perfect for busy, high-traffic sites that can't afford to have "broken" links while waiting for the new Web pages to arrive. All the replicated content is stored in a temporary directory on the server and copied to the "live" server's directories at the same time.

There are three basic types of replication in CRS: standard, automatic and incremental. These three types all deal with replication on a file-by-file basis, but vary in how they determine if a file should be updated. Standard replication performs a file-by-file synchronization. It updates or adds new files and also processes file deletions. Opting for automatic replication makes for a minimum of user intervention. As soon as the source content file is modified, the contents are replicated to the target server. Finally, incremental replication is aimed at bandwidth-conservative administrators. This mode only touches files that are newer than those sent in the last replication. This helps eliminate congesting traffic over slow lines by doing away with constant file-by-file replication.

In case things go wrong, you can even undo a replication by rolling back to a previous version. Even better, the number of replications to keep available for such a contingency is user-defined. This will bring a server back to a previous state in case a replication goes awry.

Because nothing is truly safe while in transit across the Internet, CRS makes use of NT's standard challenge/response security mechanism to guard against unauthorized replication. CRS also supports a digital signature to prevent unauthorized users from gaining access to the data. As of this beta, for a Web server to be updated using CRS, they must all be running CRS. The program will let users replicate content such as files from an FTP site or HTML documents from a non-IIS Web server to a target IIS server. In the future, Microsoft said it plans to build into CRS the ability to support non-NT/IIS servers.

Usenet newsgroups have been around as long as the Internet. Although access to these message threads is readily available to users, one thing that has not been as readily available is an easy-to-use NNTP server for Microsoft Windows NT Webmasters.

Microsoft Internet News Server

Microsoft identified this need and has developed one of the easiest-to-configure NNTP servers available, Internet News Server (INS). One word of warning: If any Internet service from MCIS is going to place a heavy load on your server, it will be this one. Though some of the other servers have the potential to increase hardware load dramatically, INS is practically guaranteed to do so. We recommend going above Microsoft's minimum hardware requirements right off the bat and saving yourself the upgrade later.

This is a good idea anyway, because Microsoft INS is only available in the full MCIS package and the company said it is not going to be available

separately. You can administer INS remotely via your Web browser and also from its property sheet found in Windows NT's Internet Service Manager.

Administrators can set Internet News Server up to poll their NNTP uplink to get the newest Usenet feed (slave mode), or INS can be the master system feeding other NNTP servers, possibly in a distributed intranet setting. INS supports Secure Sockets Layer (SSL) security, and you can control individual newsgroup access based on user names and IP addresses.

Merchant Server

Merchant Server is the flagship of Microsoft's Internet armada. It provides corporations and ISPs with a secure method of conducting business on the Internet. Merchant Server provides secure transactions both from the client and to financial institutions. When used in conjunction with IIS, users' sensitive information-such as addresses, phone numbers and credit card information-is secured by SSL. Transactional information sent to the financial institution is encrypted using Secure Electronic Transaction (SET), a protocol codeveloped by Microsoft, Visa and MasterCard. Included with Merchant Server is a trial version of VeriFone Inc.'s credit card transaction processor. Other companies are working on similar "plug-ins" for credit card handling.

Like Microsoft News Server, Merchant Server is a bit of a resource hog. This commercial-grade software requires a commercial-grade server. A P133 with 64 megabytes of RAM is the minimum recommended hardware. You also need to have an ODBC database already installed on your server for Merchant Server to set up completely. Merchant Server makes use of your ODBC database for its internal database needs. It also can read your existing product information from an ODBC database, so no data conversion is necessary.

For those who need to jump into electronic commerce as quickly as possible, Microsoft's Merchant Server comes with four sample stores you can customize or use as templates to create your own store. Merchant Server stores can make good use of ActiveX and Java applets, but they are not necessary for the store to function. Microsoft does not want to get into the payment solution provider role. The company is making the API calls available so third-party companies such as VeriFone and Cybercash can write their own payment processing applications.

Administration of your store is easy and flexible. You can update and maintain your product information from the Web and create promotions on the fly based on cross-referenced products or price. When used in conjunction with the Personalization System, your users can view the products that most interest them, and you can display other products that fall into their category of interest.

There are three main components that make up MMS: the controller, the router and the store server. The controller is the part of Merchant Server that contains the registry settings, such as language, currency and the date display format. The controller also maintains the administrator authentication settings, order-processing configurations, settings for generating buyer IDs and other functions.

Routing Shopping Requests

Merchant Server's router component handles incoming shopping requests originally received by Windows NT Server and IIS. When an online shopper makes a request over the Internet to Merchant Server, the request is first handled by IIS, then directed to the appropriate function within the Store Server. This could be anything from a request to view a page to selecting an item for purchase. Once the Store Server processes the request, it sends it back to the router, which, in turn, sends the response to the user's browser.

The Store Server handles all incoming requests that it receives from the router. In addition to supplying the user with product information and calculating the sale total, it also is responsible for post-sale support. This is primarily the ability for both the user and administrator to check on an account and order status after a purchase.

Set	Items	Description
S1	5711468	RETRIEV??? OR IDENTIF???????? OR LOCAT??? OR SEARCH??? OR Q- UERY??? OR CRAWL??? OR TRAVERS??? OR WALK???
S2	1329633	INTERNET OR INTRANET OR (INTER OR INTRA OR EXTRA)()NET OR - EXTRANET? ? OR WWW OR WEB OR PORTAL? ? OR ONLINE? OR ON()LINE?
S3	6475172	REDIRECT??? OR TRANSMIT??? OR TRANSMIS???? OR RETRANSMIT??? OR RETRANSMIS???? OR TRANSFER??? OR FORWARD??? OR ROUT??? OR MIGRAT??? OR SEND??? OR DISPATCH??? OR SENT OR RESEND??? OR - RESENT
S4	4461027	USER??? OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR BUYER? ? OR PURCHASER? ? OR CONSUMER? ? OR SHOPPER? ? OR INDIVIDUAL? ? OR PERSON? ? OR ENDUSER? ? OR VISITOR? ? OR REQUESTER? ? OR REQUESTOR? ? OR BROWSER? ? OR SURFER? ? OR SEARCHER? ? OR SPI- DER? ? OR W
S5	325	STATIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR - RESOURCE()LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ? OR SITE? ?))
S6	60732	DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR RESOURCE()LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ? OR SITE? ?)) OR AS- P? ? OR ACTIVE()SERVER()PAGE? ? OR JAF? ? OR JAVABEANS()ACTIV- ATION()FRAME
S7	3	S1 AND S2 AND S3 AND S4 AND S5 AND S6
S8	2	(S3 AND S4 AND S5 AND S6) NOT S7
S9	74	(S5 AND S6) NOT S7:S8
S10	41	RD (unique items)
S11	15	S10 AND (PY<2000 OR PD<19990427)
S12	0	AU=((ALLEN J? OR ALLEN, J?) AND (RANCK J? OR RANCK, J?))
S13	22	(AU=(ALLEN J? OR ALLEN, J? OR RANCK J? OR RANCK, J?) AND S- 6) NOT (S7:S8 OR S11)
S14	10	S13 AND (PY<2000 OR PD<19990427)
? show files		
File	2:INSPEC 1898-2006/Oct W1	(c) 2006 Institution of Electrical Engineers
File	6:NTIS 1964-2006/Oct W1	(c) 2006 NTIS, Intl Cpyrght All Rights Res
File	8:Ei Compendex(R) 1970-2006/Oct W1	(c) 2006 Elsevier Eng. Info. Inc.
File	34:SciSearch(R) Cited Ref Sci 1990-2006/Oct W1	(c) 2006 The Thomson Corp
File	35:Dissertation Abs Online 1861-2006/Sep	(c) 2006 ProQuest Info&Learning
File	56:Computer and Information Systems Abstracts 1966-2006/Sep	(c) 2006 CSA.
File	57:Electronics & Communications Abstracts 1966-2006/Sep	(c) 2006 CSA.
File	60:ANTE: Abstracts in New Tech & Engineer 1966-2006/Sep	(c) 2006 CSA.
File	65:Inside Conferences 1993-2006/Oct 11	(c) 2006 BLDSC all rts. reserv.
File	94:JICST-EPlus 1985-2006/Jul W1	(c)2006 Japan Science and Tech Corp(JST)
File	95:TEME-Technology & Management 1989-2006/Oct W2	(c) 2006 FIZ TECHNIK
File	99:Wilson Appl. Sci & Tech Abs 1983-2006/Jul	(c) 2006 The HW Wilson Co.
File	111:TGG Natl.Newspaper Index(SM) 1979-2006/Sep 27	(c) 2006 The Gale Group
File	144:Pascal 1973-2006/Sep W3	(c) 2006 INIST/CNRS
File	256:TecInfoSource 82-2006/Jan	

(c) 2006 Info.Sources Inc
File 434:SciSearch(R) Cited Ref Sci 1974-1989/Dec
(c) 2006 The Thomson Corp

1036892 USER???
 21686 VIEWER? ?
 299564 CUSTOMER? ?
 173001 CLIENT? ?
 41286 BUYER? ?
 7575 PURCHASER? ?
 290245 CONSUMER? ?
 6833 SHOPPER? ?
 1493425 INDIVIDUAL? ?
 465013 PERSON? ?
 438 ENDUSER? ?
 23780 VISITOR? ?
 995 REQUESTER? ?
 552 REQUESTOR? ?
 30030 BROWSER? ?
 2130 SURFER? ?
 5555 SEARCHER? ?
 28287 SPIDER? ?
 634 WANDERER? ?
 841038 ANT? ?
 41421 WORM? ?
 S4 4461027 USER??? OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR
 BUYER? ? OR PURCHASER? ? OR CONSUMER? ? OR SHOPPER? ? OR
 INDIVIDUAL? ? OR PERSON? ? OR ENDUSER? ? OR VISITOR? ? OR
 REQUESTER? ? OR REQUESTOR? ? OR BROWSER? ? OR SURFER? ?
 OR SEARCHER? ? OR SPIDER? ? OR WANDERER? ? OR ANT? ? OR
 WORM? ?

```

1699275 DYNAMIC
  915711 ADDRESS???
    150 HOTLINK? ?
609023 LINK? ?
  5351 URL? ?
  547146 RESOURCE
1324158 LOCAT???
  1084 RESOURCE (W) LOCAT???
    592 WEBPAGE? ?
  29734 WEBSITE? ?
    2350 HOMEPAGE? ?
  397537 HOME
  367931 WEB
  399372 PAGE? ?
  476794 DOCUMENT? ?
2287488 SITE? ?
  81495 (HOME OR WEB) (W) ((PAGE? ? OR DOCUMENT? ?) OR SITE? ?)
  2684 DYNAMIC (W) ((((((ADDRESS??? OR HOTLINK? ?) OR LINK? ?) OR
    URL? ?) OR RESOURCE (W) LOCAT???)) OR WEBPAGE? ?) OR
    WEBSITE? ?) OR HOMEPAGE? ?) ...
    57497 ASP? ?
1603692 ACTIVE
  160144 SERVER
  399372 PAGE? ?
    628 ACTIVE (W) SERVER (W) PAGE? ?
    342 JAF? ?
    1291 JAVABEANS
1366396 ACTIVATION
  804014 FRAMEWORK? ?
    4 JAVABEANS (W) ACTIVATION (W) FRAMEWORK? ?
S6 60732 DYNAMIC () (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ?
    OR RESOURCE () LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR
    HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR
    SITE? ?)) OR ASP? ? OR ACTIVE () SERVER () PAGE? ? OR JAF? ?
    OR JAVABEANS () ACTIVATION () FRAMEWORK? ?

```

11/9/14 (Item 1 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online
(c) 2006 ProQuest Info&Learning. All rts. reserv..

01650028 ORDER NO: AADNQ-27738
DESIGN AND EVALUATION OF WEB-BASED DYNAMIC HYPERTEXTS (WORLD WIDE WEB)
Author: TAM, JIM CHING
Degree: PH.D.
Year: 1997
Corporate Source/Institution: UNIVERSITY OF TORONTO (CANADA) (0779)
Adviser: MARK CHIGNELL
Source: VOLUME 59/06-B OF DISSERTATION ABSTRACTS INTERNATIONAL.
PAGE 2864. 159 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984
ISBN: 0-612-27738-0

The objective of this research is to establish the feasibility of dynamic linking as a viable complement to traditional **static links** in very large hypertexts. An experimental prototype was developed, in order to investigate the effects of dynamic linking on (1) users and (2) user performance and interaction styles during hypertext information exploration. This prototype executed queries in the background to identify appropriate targets for **dynamic links**.

Two experiments were conducted. The first experiment evaluated user performance, with either static or **dynamic links**, in terms of task time, decision time, reading time and accuracy. The second experiment studied how user performance and behaviour were affected by the visibility and modifiability of interaction feedback.

This thesis also examines the effects of dynamism on mental workload, as measured by a specially modified (hypertext) version of the Subjective Mental Workload Assessment Technique (HSWAT). Finally, the thesis considers implications for the overall relationship between information retrieval and hypertext approaches to information exploration.

The results indicated that dynamic hypertext can improve performance in question answering, particularly for domain novices or those with little computer expertise. The dynamic interface reduced workload differences between experts and novices. Link transparency had only a weak effect in improving performance in Experiment 2.

?

Microsoft is trying hard to be a one-stop shop for ISPs looking to set up and maintain an Internet presence. MCIS will help eliminate the headaches that come with managing a site consisting of a mix of different vendors' products. All the systems worked as advertised, and the tight integration with Windows NT is a definite plus.

Microsoft Commercial Internet System; Microsoft, One Microsoft Way, Redmond, Wash. 98052-6399; 206-882-8080 or fax to 206-966-7329; www.microsoft.com; price not available at press time.

Copyright 1996 CMP Media Inc.

COPYRIGHT 1996 CMP Publications Inc.

SPECIAL FEATURES: illustration; other

COMPANY NAMES: Microsoft Corp.--Products

INDUSTRY CODES/NAMES: ELEC Electronics; TELC Telecommunications; BUSN
Any type of business

DESCRIPTORS: World Wide Web--Computer programs

PRODUCT/INDUSTRY NAMES: 7372602 (Communications Software Pkgs (Micro))

SIC CODES: 7372 Prepackaged software

TICKER SYMBOLS: MSFT

TRADE NAMES: Microsoft Commercial Internet Systems (Internet/Web server software)--Evaluation

FILE SEGMENT: CD File 275

Set	Items	Description
S1	1980131	RETRIEV??? OR IDENTIF??????? OR LOCAT??? OR SEARCH??? OR Q- UERY??? OR CRAWL??? OR TRAVERS??? OR WALK???
S2	338258	INTERNET OR INTRANET OR (INTER OR INTRA OR EXTRA) () NET OR - EXTRANET? ? OR WWW OR WEB OR PORTAL? ? OR ONLINE? OR ON() LINE?
S3	3713345	REDIRECT??? OR TRANSMIT??? OR TRANSMIS???? OR RETRANSMIT??? OR RETRANSMIS???? OR TRANSFER??? OR FORWARD??? OR ROUT??? OR MIGRAT??? OR SEND??? OR DISPATCH??? OR SENT OR RESEND??? OR - RESENT
S4	1818004	USER??? OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR BUYER? ? OR PURCHASER? ? OR CONSUMER? ? OR SHOPPER? ? OR INDIVIDUAL? ? OR PERSON? ? OR ENDUSER? ? OR VISITOR? ? OR REQUESTER? ? OR REQUESTOR? ? OR BROWSER? ? OR SURFER? ? OR SEARCHER? ? OR SPI- DER? ? OR W
S5	165	STATIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR - RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR SITE? ?))
S6	14291	DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR SITE? ?)) OR AS- P? ? OR ACTIVE() SERVER() PAGE? ? OR JAF? ? OR JAVABEANS() ACTIV- ATION() FRAME
S7	6	S1 AND S2 AND S3 AND S4 AND S5 AND S6
S8	6	(S3 AND S4 AND S5 AND S6) NOT S7
S9	44	(S5 AND S6) NOT S7:S8
S10	3	AU=((ALLEN J? OR ALLEN, J?) AND (RANCK J? OR RANCK, J?))
S11	1	(S9 AND (MC=T01 OR IC=(G06F-015/16 OR G06F-007/00))) NOT (- S7:S8 OR S10 OR AD=(19990427:20020427) OR AD=(20020428:200610- 11))

? show files

File 347:JAPIO Dec 1976-2006/Jan(Updated 061009)

(c) 2006 JPO & JAPIO

File 350:Derwent WPIX 1963-2006/UD=200665

(c) 2006 The Thomson Corporation

?

722298 USER???
 20955 VIEWER? ?
 80028 CUSTOMER? ?
 79992 CLIENT? ?
 6219 BUYER? ?
 7905 PURCHASER? ?
 44424 CONSUMER? ?
 1033 SHOPPER? ?
 349937 INDIVIDUAL? ?
 267835 PERSON? ?
 31 ENDUSER? ?
 7656 VISITOR? ?
 3771 REQUESTER? ?
 1346 REQUESTOR? ?
 17294 BROWSER? ?
 700 SURFER? ?
 2199 SEARCHER? ?
 7002 SPIDER? ?
 56 WANDERER? ?
 389217 ANT? ?
 33965 WORM? ?
 S4 1818004 USER??? OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR
 BUYER? ? OR PURCHASER? ? OR CONSUMER? ? OR SHOPPER? ? OR
 INDIVIDUAL? ? OR PERSON? ? OR ENDUSER? ? OR VISITOR? ? OR
 REQUESTER? ? OR REQUESTOR? ? OR BROWSER? ? OR SURFER? ?
 OR SEARCHER? ? OR SPIDER? ? OR WANDERER? ? OR ANT? ? OR
 WORM? ?

```

185022 DYNAMIC
    312397 ADDRESS???
        17 HOTLINK? ?
    349469 LINK? ?
        8853 URL? ?
        44863 RESOURCE
1211594 LOCAT???
    3754 RESOURCE (W) LOCAT???
    1448 WEBPAGE? ?
    5952 WEBSITE? ?
    4603 HOMEPAGE? ?
    88225 HOME
    169176 WEB
    108886 PAGE? ?
    143525 DOCUMENT? ?
    215289 SITE? ?
    26548 (HOME OR WEB) (W) ((PAGE? ? OR DOCUMENT? ?) OR SITE? ?)
    1147 DYNAMIC (W) ((((((ADDRESS??? OR HOTLINK? ?) OR LINK? ?) OR
        URL? ?) OR RESOURCE (W) LOCAT???) OR WEBPAGE? ?) OR
        WEBSITE? ?) OR HOMEPAGE? ?) ...
    13100 ASP? ?
    667740 ACTIVE
    209042 SERVER
    108886 PAGE? ?
        134 ACTIVE (W) SERVER (W) PAGE? ?
        19 JAF? ?
        79 JAVABEANS
    91645 ACTIVATION
    43811 FRAMEWORK? ?
        0 JAVABEANS (W) ACTIVATION (W) FRAMEWORK? ?
S6 14291 DYNAMIC () (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ?
    OR RESOURCE () LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR
    HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR
    SITE? ?)) OR ASP? ? OR ACTIVE () SERVER () PAGE? ? OR JAF? ?
    OR JAVABEANS () ACTIVATION () FRAMEWORK? ?

```

10/5/3 (Item 3 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2006 The Thomson Corporation. All rts. reserv.

0014725934 - Drawing available
WPI ACC NO: 2005-073553/200508
Related WPI Acc No: 2005-313790; 2005-313866
XRPX Acc No: N2005-063480

Static address to dynamic address conversion providing method for e.g.
Internet, involves generating dynamic address incorporating values
associated with fields, where dynamic address points to dynamic Web page
Patent Assignee: MICROSOFT CORP (MICT)
Inventor: ALLEN J ; RANCK J L
Patent Family (1 patents, 1 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
US 20040260722	A1	20041223	US 2000560703	A	20000427	200508 B
			US 2004893006	A	20040716	

Priority Applications (no., kind, date): US 2000560703 A 20000427; US
2004893006 A 20040716

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
US 20040260722	A1	EN	15	5	Continuation of application US 2000560703

Alerting Abstract US A1

NOVELTY - The method involves receiving a static address (172) pointing
to a dynamic Web page, and parsing the static address to identify values
associated with a field within the static address. A dynamic address (112)
incorporating the values associated with fields, is generated, where the
dynamic address points to the dynamic Web page. The dynamic address is
formed by hostname (144), path (146) and searchvalues fields.

DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

- 1.a computer-readable storage media having computer-executable
instructions that, when executed by a computer, performs a static
address to dynamic address conversion method
- 2.a dynamic to static web address conversion method
- 3.a computer-readable storage media having computer-executable
instructions that, when executed by a computer, performs a dynamic to
static web address conversion method.

USE - Used for providing a static address to dynamic address (S-to-D
address) conversion in a computer network e.g. Internet.

ADVANTAGE - The method allows the spiders to fetch and indexers to index
the dynamic content of dynamic Web sites. The method allows the browser
utilizing a static address of a dynamic Web site, to access the referenced
dynamic Web page, instead of a stale, static copy that does not give
accurate information.

DESCRIPTION OF DRAWINGS - The drawing shows a bi-directional
address-conversion-mapping diagram with an implementation of
dynamic-to-static (D-to-S) address conversion from top to bottom, and an
implementation of static-to-dynamic (S-to-D) address conversion from bottom
to top.

- 100 Address conversion mapping
- 112 Dynamic address
- 144 Hostname field

146 Path field
172 Static address

Title Terms/Index Terms/Additional Words: STATIC; ADDRESS; DYNAMIC; CONVERT
; METHOD; GENERATE; INCORPORATE; VALUE; ASSOCIATE; FIELD; POINT; WEB;
PAGE

Class Codes

International Classification (Main): G06F-017/00
US Classification, Issued: 707102000
File Segment: EPI;
DWPI Class: T01
Manual Codes (EPI/S-X): T01-J11A1; T01-N02A1A; T01-S03

Set	Items	Description
S1	14291	DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR RESOURCE()LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR SITE? ?)) OR ASP? ? OR ACTIVE() SERVER() PAGE? ? OR JAF? ? OR JAVABEANS() ACTIVATION() FRAME
S2	4	(AU=(ALLEN J? OR ALLEN, J? OR RANCK J? OR RANCK, J?) AND S-1) NOT (AD=(19990427:20020427) OR AD=(20020428:20061011))

? show files

File 347:JAPIO Dec 1976-2006/Jan(Updated 061009)
(c) 2006 JPO & JAPIO

File 350:Derwent WPIX 1963-2006/UD=200665
(c) 2006 The Thomson Corporation .

```

185022 DYNAMIC
312397 ADDRESS???
    17 HOTLINK? ?
349469 LINK? ?
    8853 URL? ?
    44863 RESOURCE
1211594 LOCAT???
    3754 RESOURCE (W) LOCAT???
    1448 WEBPAGE? ?
    5952 WEBSITE? ?
    4603 HOMEPAGE? ?
    88225 HOME
169176 WEB
108886 PAGE? ?
143525 DOCUMENT? ?
215289 SITE? ?
    26548 (HOME OR WEB) (W) ((PAGE? ? OR DOCUMENT? ?) OR SITE? ?)
    1147 DYNAMIC (W) ((((((ADDRESS??? OR HOTLINK? ?) OR LINK? ?) OR
        URL? ?) OR RESOURCE (W) LOCAT???) OR WEBPAGE? ?) OR
        WEBSITE? ?) OR HOMEPAGE? ?) ...
    13100 ASP? ?
667740 ACTIVE
209042 SERVER
108886 PAGE? ?
    134 ACTIVE (W) SERVER (W) PAGE? ?
    19 JAF? ?
    79 JAVABEANS
91645 ACTIVATION
43811 FRAMEWORK? ?
    0 JAVABEANS (W) ACTIVATION (W) FRAMEWORK? ?
S1 14291 DYNAMIC () (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ?
        OR RESOURCE () LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR
        HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR
        SITE? ?)) OR ASP? ? OR ACTIVE () SERVER () PAGE? ? OR JAF? ?
        OR JAVABEANS () ACTIVATION () FRAMEWORK? ?

```

Set	Items	Description
S1	52245	(RETRIEV??? OR IDENTIF??????? OR LOCAT??? OR SEARCH??? OR - QUERY??? OR CRAWL??? OR TRAVERS??? OR WALK???) (10N) (INTERNET - OR INTRANET OR (INTER OR INTRA OR EXTRA) () NET OR EXTRANET? ? - OR WWW OR WEB OR PORTAL? ? OR ONLINE? OR ON() LINE?)
S2	140108	(REDIRECT??? OR TRANSMIT??? OR TRANSMIS???? OR RETRANSMIT?- ?? OR RETRANSMIS???? OR TRANSFER??? OR FORWARD??? OR ROUT??? - OR MIGRAT??? OR SEND??? OR DISPATCH??? OR SENT OR RESEND??? - OR RESENT) (5N) (USER??? OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR BUYER? ?
S3	11	S2 (5N) (STATIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOME- PAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR SITE? ?) -))
S4	0	S3 (5N) (DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR UR- L? ? OR RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HO- MEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR SITE? - ?)) OR ASP? ? OR ACTIVE() SERVER() PAGE? ? OR JAF? ? OR JAVABEA- NS() ACTIVATION
S5	1	S3 NOT (PD=(19990427:20020427) OR PD=(20020428:20061011))
S6	868	STATIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR - RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR SITE? ?))
S7	48548	DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ? OR RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR SITE? ?)) OR AS- P? ? OR ACTIVE() SERVER() PAGE? ? OR JAF? ? OR JAVABEANS() ACTIV- ATION() FRAME
S8	0	S5 AND S7
S9	404	S2 AND S6 AND S7
S10	21	S9 AND IC=(G06F-015/16 OR G06F-007/00)
S11	4	S10 NOT (PD=(19990427:20020427) OR PD=(20020428:20061011))
S12	0	(S2(10N)S6(10N)S7) NOT (S3 OR S5 OR S11 OR PD=(19990427:20- 020427) OR PD=(20020428:20061011))
S13	1	(S2(100N)S6(100N)S7) NOT (S3 OR S5 OR S11 OR PD=(19990427:- 20020427) OR PD=(20020428:20061011))
? show files		
File 348:EUROPEAN PATENTS 1978-2006/ 200640		
(c) 2006 European Patent Office		
File 349:PCT FULLTEXT 1979-2006/UB=20061005UT=20060928		
(c) 2006 WIPO/Thomson		
?		

26269 REDIRECT???
 459738 TRANSMIT???
 472646 TRANSMIS????
 11770 RETRANSMIT???
 12163 RETRANSMIS????
 876415 TRANSFER???
 293583 FORWARD???
 335624 ROUT???
 99402 MIGRAT???
 251346 SEND???
 424601 DISPATCH???
 214317 SENT
 3700 RESEND???
 5900 RESENT
 427166 USER???
 30406 VIEWER? ?
 72974 CUSTOMER? ?
 62969 CLIENT? ?
 6671 BUYER? ?
 9489 PURCHASER? ?
 90447 CONSUMER? ?
 1840 SHOPPER? ?
 557656 INDIVIDUAL? ?
 377900 PERSON? ?
 964 ENDUSER? ?
 7159 VISITOR? ?
 4341 REQUESTER? ?
 3784 REQUESTOR? ?
 32934 BROWSER? ?
 810 SURFER? ?
 2638 SEARCHER? ?
 7236 SPIDER? ?
 134 WANDERER? ?
 379381 ANT? ?
 19905 WORM? ?
 S2 140108 (REDIRECT??? OR TRANSMIT??? OR TRANSMIS???? OR
 RETRANSMIT??? OR RETRANSMIS???? OR TRANSFER??? OR
 FORWARD??? OR ROUT??? OR MIGRAT??? OR SEND??? OR
 DISPATCH??? OR SENT OR RESEND??? OR RESENT) (5N) (USER???
 OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR BUYER? ? OR
 PURCHASER? ? OR CONSUMER? ? OR SHOPPER? ? OR INDIVIDUAL?
 ? OR PERSON? ? OR ENDUSER? ? OR VISITOR? ? OR REQUESTER?
 ? OR REQUESTOR? ? OR BROWSER? ? OR SURFER? ? OR SEARCHER?
 ? OR SPIDER? ? OR WANDERER? ? OR ANT? ? OR WORM? ?)

11 S3

```
160679 DYNAMIC
355010 ADDRESS???
    164 HOTLINK? ?
290764 LINK? ?
    22075 URL? ?
    56247 RESOURCE
887399 LOCAT???
    10554 RESOURCE(W) LOCAT???
    1938 WEBPAGE? ?
    15161 WEBSITE? ?
    2327 HOMEPAGE? ?
    94850 HOME
140023 WEB
302576 PAGE? ?
483248 DOCUMENT? ?
342207 SITE? ?
    36188 (HOME OR WEB)(W)((PAGE? ? OR DOCUMENT? ?) OR SITE? ?)
    3434 DYNAMIC(W)((((ADDRESS??? OR HOTLINK? ?) OR LINK? ?) OR
        URL? ?) OR RESOURCE(W)LOCAT???) OR WEBPAGE? ?) OR
        WEBSITE? ?) OR HOMEPAGE? ?)...
    44907 ASP? ?
547212 ACTIVE
    96689 SERVER
302576 PAGE? ?
    1184 ACTIVE(W) SERVER(W) PAGE? ?
    400 JAF? ?
    591 JAVABEANS
194687 ACTIVATION
    52652 FRAMEWORK? ?
        2 JAVABEANS(W) ACTIVATION(W) FRAMEWORK? ?
S4      0 S3(5N) (DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR
        URL? ? OR RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ?
        OR HOMEPAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ?
        OR SITE? ?)) OR ASP? ? OR ACTIVE() SERVER() PAGE? ? OR JAF?
        ? OR JAVABEANS() ACTIVATION() FRAMEWORK? ?)
```

```

160679 DYNAMIC
    355010 ADDRESS???
    164 HOTLINK? ?
    290764 LINK? ?
    22075 URL? ?
    56247 RESOURCE
    887399 LOCAT???
    10554 RESOURCE (W) LOCAT???
    1938 WEBPAGE? ?
    15161 WEBSITE? ?
    2327 HOMEPAGE? ?
    94850 HOME
    140023 WEB
    302576 PAGE? ?
    483248 DOCUMENT? ?
    342207 SITE? ?
    36188 (HOME OR WEB) (W) ((PAGE? ? OR DOCUMENT? ?) OR SITE? ?)
    3434 DYNAMIC (W) ((((((ADDRESS??? OR HOTLINK? ?) OR LINK? ?) OR
        URL? ?) OR RESOURCE (W) LOCAT???) OR WEBPAGE? ?) OR
        WEBSITE? ?) OR HOMEPAGE? ?) ...
    44907 ASP? ?
    547212 ACTIVE
    96689 SERVER
    302576 PAGE? ?
    1184 ACTIVE (W) SERVER (W) PAGE? ?
    400 JAF? ?
    591 JAVABEANS
    194687 ACTIVATION
    52652 FRAMEWORK? ?
    2 JAVABEANS (W) ACTIVATION (W) FRAMEWORK? ?
S7 48548 DYNAMIC () (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ?
    OR RESOURCE () LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR
    HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR
    SITE? ?)) OR ASP? ? OR ACTIVE () SERVER () PAGE? ? OR JAF? ?
    OR JAVABEANS () ACTIVATION () FRAMEWORK? ?

```

140246 REDIRECT???
 819933 TRANSMIT???
 1605861 TRANSMIS????
 11267 RETRANSMIT???
 21421 RETRANSMIS????
 2555376 TRANSFER???
 6826927 FORWARD???
 2370618 ROUT???
 774022 MIGRAT???
 2343857 SEND???
 290961 DISPATCH???
 2053622 SENT
 6497 RESEND???
 33662 RESENT
 6782171 USER???
 618649 VIEWER? ?
 11930529 CUSTOMER? ?
 4757391 CLIENT? ?
 1926018 BUYER? ?
 335527 PURCHASER? ?
 7035273 CONSUMER? ?
 463585 SHOPPER? ?
 4986695 INDIVIDUAL? ?
 2898893 PERSON? ?
 7606 ENDUSER? ?
 958314 VISITOR? ?
 9900 REQUESTER? ?
 4154 REQUESTOR? ?
 657177 BROWSER? ?
 83795 SURFER? ?
 36251 SEARCHER? ?
 80643 SPIDER? ?
 10335 WANDERER? ?
 1565895 ANT? ?
 131747 WORM? ?
 S2 1060855 (REDIRECT??? OR TRANSMIT??? OR TRANSMIS???? OR
 RETRANSMIT??? OR RETRANSMIS???? OR TRANSFER??? OR
 FORWARD??? OR ROUT??? OR MIGRAT??? OR SEND??? OR
 DISPATCH??? OR SENT OR RESEND??? OR RESENT) (5N) (USER???
 OR VIEWER? ? OR CUSTOMER? ? OR CLIENT? ? OR BUYER? ? OR
 PURCHASER? ? OR CONSUMER? ? OR SHOPPER? ? OR INDIVIDUAL?
 ? OR PERSON? ? OR ENDUSER? ? OR VISITOR? ? OR REQUESTER?
 ? OR REQUESTOR? ? OR BROWSER? ? OR SURFER? ? OR SEARCHER?
 ? OR SPIDER? ? OR WANDERER? ? OR ANT? ? OR WORM? ?)

15 S3

```
1286414 DYNAMIC
5080254 ADDRESS???
    7756 HOTLINK? ?
2830376 LINK? ?
1229196 URL? ?
1937061 RESOURCE
6071455 LOCAT???
    9366 RESOURCE(W) LOCAT???
    11643 WEBPAGE? ?
2537505 WEBSITE? ?
    87632 HOMEPAGE? ?
    7128963 HOME
10163212 WEB
    4066160 PAGE? ?
    2652154 DOCUMENT? ?
11180071 SITE? ?
    7677234 (HOME OR WEB)(W)((PAGE? ? OR DOCUMENT? ?) OR SITE? ?)
    25410 DYNAMIC(W)((((ADDRESS??? OR HOTLINK? ?) OR LINK? ?) OR
        URL? ?) OR RESOURCE(W)LOCAT??? OR WEBPAGE? ?) OR
        WEBSITE? ?) OR HOMEPAGE? ?)...
    502252 ASP? ?
2459163 ACTIVE
1948999 SERVER
4066160 PAGE? ?
    10394 ACTIVE(W) SERVER(W) PAGE? ?
    1896 JAF? ?
    19433 JAVABEANS
    203289 ACTIVATION
1009283 FRAMEWORK? ?
    25 JAVABEANS(W) ACTIVATION(W) FRAMEWORK? ?
S4      0 S3(5N) (DYNAMIC() (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR
        URL? ? OR RESOURCE() LOCAT??? OR WEBPAGE? ? OR WEBSITE? ?
        OR HOMEPAGE? ? OR (HOME OR WEB)() (PAGE? ? OR DOCUMENT? ?
        OR SITE? ?)) OR ASP? ? OR ACTIVE() SERVER() PAGE? ? OR JAF?
        ? OR JAVABEANS() ACTIVATION() FRAMEWORK? ?)
```

```

1286414 DYNAMIC
5080254 ADDRESS???
7756 HOTLINK? ?
2830376 LINK? ?
1229196 URL? ?
1937061 RESOURCE
6071455 LOCAT???
9366 RESOURCE (W) LOCAT???
11643 WEBPAGE? ?
2537505 WEBSITE? ?
87632 HOMEPAGE? ?
7128963 HOME
10163212 WEB
4066160 PAGE? ?
2652154 DOCUMENT? ?
11180071 SITE? ?
7677234 (HOME OR WEB) (W) ((PAGE? ? OR DOCUMENT? ?) OR SITE? ?)
25410 DYNAMIC (W) ((((((ADDRESS??? OR HOTLINK? ?) OR LINK? ?) OR
URL? ?) OR RESOURCE (W) LOCAT???)) OR WEBPAGE? ?) OR
WEBSITE? ?) OR HOMEPAGE? ?) ...
502252 ASP? ?
2459163 ACTIVE
1948999 SERVER
4066160 PAGE? ?
10394 ACTIVE (W) SERVER (W) PAGE? ?
1896 JAF? ?
19433 JAVABEANS
203289 ACTIVATION
1009283 FRAMEWORK? ?
25 JAVABEANS (W) ACTIVATION (W) FRAMEWORK? ?
S6 533347 DYNAMIC () (ADDRESS??? OR HOTLINK? ? OR LINK? ? OR URL? ?
OR RESOURCE () LOCAT??? OR WEBPAGE? ? OR WEBSITE? ? OR
HOMEPAGE? ? OR (HOME OR WEB) () (PAGE? ? OR DOCUMENT? ? OR
SITE? ?)) OR ASP? ? OR ACTIVE () SERVER () PAGE? ? OR JAF? ?
OR JAVABEANS () ACTIVATION () FRAMEWORK? ?

```